



Universitat
Oberta
de Catalunya

Adaptació d'Odoo per a una empresa fustera

Estudiant: Manuel Regidor González
Estudis: Grau Enginyeria Informàtica
Àrea: Enginyeria del software

Consultor: Oriol Martí Girona
Professor responsable de l'assignatura: Santi Caballe Llobet

Lliurament: 01/2021

Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](#)

Títol del treball:	<i>Adaptació d'Odoo per a una empresa fustera</i>
Nom de l'autor:	Manuel Regidor González
Nom del consultor:	Oriol Martí Girona
Data de lliurament (mm/aaaa):	<i>01/2021</i>
Àrea del Treball Final:	Enginyeria del software
Titulació:	<i>Grau d'Enginyeria Informàtica</i>
Resum:	
<p>Aquesta memòria recull part del procés de desenvolupament d'una sèrie de mòduls que adapten l'ERP Odoo a les necessitats d'una empresa fustera. Concretament, el client necessita enregistrar la fusta emprada pels productes que ven, calcular els preus a partir de les seues dimensions, aplicar tractaments i indicar l'ús de fusta sostenible. Tota aquesta informació s'ha de traslladar, a més a més, als informes PDF generats pel software.</p> <p>L'especificitat dels requeriments expressats pel client són inabastables en la seua totalitat a través de les funcionalitats oferides per Odoo. És per això necessari analitzar amb profunditat de quina manera es poden fer servir els recursos disponibles, com s'ha d'utilitzar l'herència perquè aporten noves opcions i quins nous elements s'han de crear.</p> <p>Aquesta memòria parteix de la valoració de la viabilitat tècnica del projecte i es documenten l'especificació de requisits, la fase de disseny amb la definició precisa de les classes, mètodes i vistes i la planificació de l'etapa de proves. El disseny ha estat guiat pel paradigma del Desenvolupament Guiat per Proves (TDD).</p>	
Abstract:	
<p>This report covers part of the process of the development of a series of modules which adapt the ERP Odoo to the needs of a wood company. In particular, the client needs to register the wood used in the products, calculate their prices through their dimensions, apply treatments, and indicate the use of sustainable wood.</p> <p>The specificity of the client's requirements cannot be entirely fulfilled by the functionalities offered by Odoo. That is why it is necessary to deeply analyse how the available resources can be employed, how inheritance can be used to provide new options, and which elements need to be created.</p> <p>This report starts with the assessment of the technical viability of the project and the documentation includes the requirements specification, the design phase with the precise definition of classes, methods, and views, and the planning of the testing stage. The design has followed the Test Driven Development (TDD) principles.</p>	
Paraules clau (entre 4 i 8):	
Odoo, ERP, TDD	

ÍNDEX

1. Definició del projecte i context	5
1.1 Sobre l'ERP Odoo	6
1.2 L'ús d'Odoo en el context d'aquest treball	7
2. Planificació	8
3. Principals opcions de disseny	9
4. Requisits	11
4.1. Introducció	11
4.1.1. Propòsit	11
4.1.2. Abast	11
4.1.3. Acrònims i abreviatures	12
4.1.4. Resum	12
4.2. Descripció general	13
4.2.1. Perspectiva del producte	13
4.2.1.1. Productes	13
4.2.1.2. Vendes	15
4.2.1.3. Albarans	16
4.2.1.4. Factures	17
4.2.1.5. Preus i Pricelists	18
4.2.2. Funcionalitats del producte	19
4.2.2.1. Tipologia de productes	19
4.2.2.2. Càlcul de fusta emprada i volum, superfície o longitud dels PG	20
4.2.2.3. Càlcul dels preus	22
4.2.2.4. Vendes, factures i albarans amb fusta PEFC	22
4.2.2.5. Aplicació de tractaments	23
4.2.2.6. Generació de documents amb nova informació	23
4.2.2.7. Dependències entre funcionalitats	24
4.2.3. Característiques de l'usuari	25
4.2.4. Restriccions de desenvolupament	25
4.2.4.1. Restriccions funcionals	25
4.2.4.2. Restriccions del llenguatge	26
4.2.4.3. Restriccions d'interfície gràfica	27
4.2.4.4. Restriccions de seguretat	27
4.2.5. Suposicions i dependències	27
4.3. Requisits específics	28

4.3.1. Requisits d'interfície externa	28
4.3.1.1. Interfície d'usuari	28
4.3.2. Característiques del sistema	33
4.3.2.1. Tipologia de productes	34
4.3.2.2. Càlcul de la fusta emprada i del volum, superfície o longitud dels PG	35
4.3.2.3. Càlcul dels preus dels PG	36
4.3.2.4. Vendes, factures i albarans amb fusta PEFC	36
4.3.2.5. Aplicació de tractaments als productes	37
4.3.2.6. Generació de documents amb nova informació	38
4.3.3. Requisits de rendiment	38
4.3.4. Restriccions de disseny	39
4.3.5. Atributs del sistema de software	39
4.3.6. Altres requisits no funcionals	39
4.4. Producte mínim lliurable	40
4.5. Tests d'acceptació	41
5. Disseny	63
5.1. Arquitectura d'Odoo	63
5.2. Mòduls a Odoo	64
5.2.1. Objectes de negoci	64
5.2.1.1. Camps relacionals	65
5.2.1.2. Camps computats	67
5.2.1.3. Restriccions	68
5.2.1.4. Seguretat	68
5.2.1.5. Wizards	69
5.2.1.6. Herències	69
5.2.2. Vistes	70
5.2.2.1. Herències	71
5.2.3. Templates	72
5.3. Distribució del projecte en mòduls	72
5.4. Classes, vistes, mètodes, informes i data dels mòduls	74
5.4.1. Models heretats	74
5.4.1.1. product.template i product.product	74
5.4.1.2. sale.order i sale.order.line	75
5.4.1.3. stock.picking, stock.move i stock.move.line	75
5.4.1.4. account.move i account.move.line	76
5.4.2. Generic Products and Used Wood	77
5.4.2.1. Atributs	77
5.4.2.2. Operacions i restriccions	83
5.4.2.3. Vistes	92
5.4.2.4. Data	94

5.4.3. Generic Products Price	95
5.4.3.1. Atributs	95
5.4.3.2. Operacions i restriccions	97
5.4.3.3. Vistes	100
5.4.4. PEFC Instances	102
5.4.4.1. Atributs	102
5.4.4.2. Operacions i restriccions	103
5.4.4.3. Vistes	103
5.4.5. Treatments Application	104
5.4.5.1. Atributs	104
5.4.5.2. Operacions i restriccions	107
5.4.5.3. Vistes	110
5.4.6. New Information Reports	112
5.4.6.1. Atributs	112
5.4.6.2. Operacions i restriccions	114
5.4.6.3. Vistes	115
5.4.6.4. Informes	115
6. Implementació i proves	117
6.1. Implementació	117
6.2. Proves	118
6.2.1. Planificació de les proves	118
6.2.2. Definició de proves concretes	119
6.2.2.1. Nivell de les proves	119
6.2.2.2. Tècniques de proves	120
7. Conclusions i continuïtat	121
Annexos	123
1. Flux bàsic d'Odoo	123
2. Casos d'ús	124
Bibliografia	132

1. Definició del projecte i context

La idea del TFG parteix d'un cas real en què un client, una empresa que fabrica i ven productes de fusta (a partir d'ara 'Client'), requereix una personalització profunda de l'ERP Odoo per adaptar-lo a les seues necessitats. Les particularitats del funcionament d'aquest client són bàsicament 6:

- Existeixen una sèrie de **productes genèrics** les dimensions dels quals són configurables. Per exemple, s'utilitza el producte genèric 'Biga' les dimensions del qual són variables. Podem tindre una biga de 2m x 20cm x 20cm i una altra de 2.3m x 22cm x 22cm. En els dos casos es tracta amb el producte 'Biga,' però cadascuna té unes característiques diferents i un volum de fusta distint. El preu haurà de poder ser calculat a partir d'aquestes dimensions i de la unitat secundària de cada producte, que pot ser metres cúbics, metres quadrats o metres lineals.
- Existeixen productes en què es pot **introduir manualment el volum de fusta** que representen. Per exemple, en una venda, el producte 'Llenya' pot representar un volum de fusta X i en una altra venda el producte 'Llenya' pot representar un volum de fusta Y.
- Existeixen productes en què el volum de fusta que representen es calcula a partir de la seua quantitat mitjançant un **coeficient**. Per exemple, en el producte 'Serradures' el volum de fusta es calcula multiplicant el seu pes en kilograms per 1.15.
- Necessitat de portar un registre de la **fusta venuda** a partir de tots aquests productes.
- Alguns productes poden acceptar **tractaments** com, per exemple, pintura o poliment.
- S'ha d'indicar quan la fusta emprada als productes porta la certificació **PEFC**, que indica l'explotació sostenible dels recursos forestals.

Independentment de les solucions finals, i malgrat la necessitat de realitzar una personalització prou ampla, amb el coneixement del *software* i la definició dels requisits bàsics anteriors **sembla viable tècnicament realitzar aquest desenvolupament**.

L'**objectiu** és, per tant, seguir els diferents passos del procés de desenvolupament de *software* per a aconseguir adaptar Odoo a les exigències esmentades. El projecte hauria de finalitzar amb el desenvolupament d'una sèrie de mòduls que aconseguen totes eixes funcionalitats. En el moment de lliurar aquesta memòria, s'havien obtingut els següents **elements**:

- Una relació exhaustiva dels requisits, documentada segons l'estàndar *EEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications* de la Computer Society.
- Una sèrie de tests, emprats tant per a realitzar proves com per guiar la fase de disseny.

- Esquemes invariants de cadascuna de les entitats dels mòduls, així com la definició dels mètodes necessaris, les restriccions i les vistes.
- Una planificació d'un pla de proves.
- La versió beta de dos dels mòduls.

1.1 Sobre l'ERP Odoo

Odoo és un sistema de planificació de recursos empresarials (**ERP**). Aquest *software* permet gestionar diferents aspectes dins d'una empresa, entre els quals es troben les vendes, els albarans, la facturació, l'inventari, la tenda electrònica, el lloc web i la comptabilitat. Odoo compta amb una versió gratuïta anomenada 'Community' i amb una versió de pagament anomenada 'Enterprise' i les possibilitats i funcionalitats de la segona són més àmplies que les de la primera.

El *software* es pot adaptar a les necessitats dels clients de diferents maneres:

- Amb la instal·lació de **mòduls base**. Odoo disposa d'una sèrie de mòduls base que es corresponen amb diferents aspectes de la gestió empresarial. Així, no cal instal·lar tot el *software* en el seu conjunt sinó només aquells mòduls que el client farà servir per a la seua activitat. Això vol dir que, per exemple, si una empresa treballa l'inventari amb Odoo però la comptabilitat està gestionada per una assessoria externa no requerirà que tinga el mòdul de comptabilitat instal·lat. Alguns d'aquests mòduls només estan disponibles per a la versió Enterprise de l'ERP, mentre que d'altres estan disponibles per a qualsevol usuari, independentment que tinguen la versió gratuïta o la versió de pagament.
- Amb la instal·lació de mòduls procedents de la comunitat. **OCA** (Odoo Community Association) és una organització que facilita l'intercanvi de desenvolupaments per a Odoo. Disposa d'un repositori a Github des del qual es poden descarregar mòduls de manera gratuïta desenvolupats i aprovats per una àmplia comunitat de desenvolupadors.
- Amb la **compra de mòduls**. Existeixen empreses que es dediquen a la comercialització de mòduls per a Odoo.
- Amb **desenvolupaments customitzats**. De vegades, les necessitats dels clients hi poden anar més enllà dels mòduls oferits de base, dels que es poden trobar al mercat i de les funcionalitats disponibles al repositori OCA.

1.2 L'ús d'Odoo en el context d'aquest treball

De les diferents opcions d'adaptació presentades a l'apartat anterior, en el context de la petició realitzada pel client sobre el qual es basa aquest TFG, és necessari realitzar un **desenvolupament customitzat**. Entre els motius més directes es troben els següents:

- Existeixen una sèrie de productes amb comportaments diferents dels oferits de base per Odoo, tant pel que fa al càlcul de preus a partir d'unitats de mesura secundàries com a l'aplicació de tractaments.
- Cal realitzar un enregistrament de la fusta utilitzada pels productes, un aspecte molt específic que ni Odoo, ni empreses desenvolupadores ni la comunitat contemplen.

L'existència d'aquests punts afecta, a més a més, al flux bàsic de treball en Odoo.



1. La **venda** (Sale) inclou els productes que el client desitja adquirir.
2. Quan s'accepta la venda, es genera un **albarà** (Picking), que representa un moviment a l'inventari.
3. Una vegada s'accepta la venda, es pot generar una **factura** (Invoice).

Ja des de la primera etapa (generació de la venda) s'hauran de considerar les característiques d'aquests productes especials i, a més a més, aquesta informació s'haurà de transferir d'un pas cap al següent. És a dir, si la venda inclou un producte Biga de 2m x 20cm x 20cm, aquesta informació de dimensions s'haurà de traspasar tant a l'albarà com a la factura, per disposar d'aquestes dades durant tot el cicle.

Així i tot, donat que es disposa d'un marc sobre el qual treballar, s'haurà d'intentar utilitzar les opcions que ofereix Odoo de base quan siga possible. Entre les raons per a optar per aquesta política estan les següents:

- Estalviar temps.
- Utilitzar un sistema testat i sota supervisió permanent.
- Evitar afectar per error el funcionament d'altres parts del sistema que estiguen relacionades.

2. Planificació

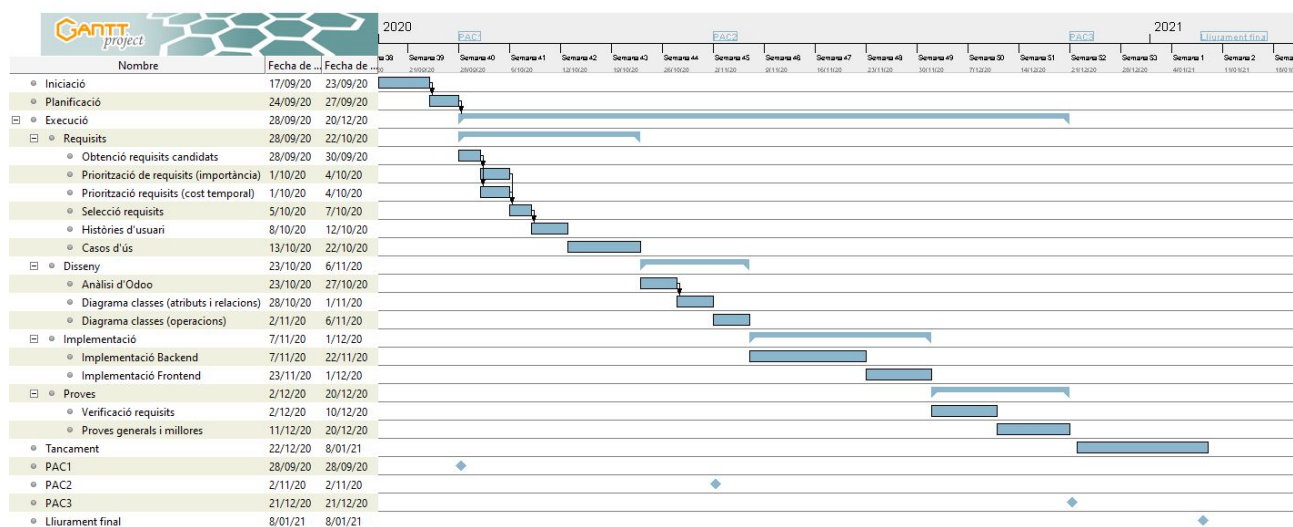
A l'etapa d'iniciació del projecte es va definir, com s'indica al mòdul "La gestió de projectes. Conceptes bàsics" de l'assignatura "Gestió de projectes", la conceptualització i definició del projecte amb la identificació dels requisits bàsics i es va confirmar la viabilitat tècnica¹.

Com a part de la fase de planificació es va dissenyar un diagrama de Gantt amb una estimació temporal de les tasques generals per a completar el TFG. Pel tipus de projecte, aquest TFG para especial atenció a tres aspectes:

- L'ús de **patrons disponibles a Odoo** aplicables a aquest desenvolupament concret. Aquest aspecte es té en compte en tota la fase de disseny i implementació.
- **Recull de requisits** i la seua documentació detallada.
- **Fase de test**. Assegurar que els requisits recollits es compleixen en el desenvolupament.

Així mateix, s'ha realitzat una part de la implementació.

La planificació es basa en metodologia clàssica i s'ha utilitzat el cicle de vida del *software* present al mòdul 2 de l'assignatura 'Projecte de desenvolupament de programari', en què s'estableixen cinc fases: Requisits, Disseny, Implementació, Proves i Manteniment² (aquesta última no s'ha desenvolupat en aquest treball).



¹ RODRÍGUEZ, Juan Ramón. *La gestió de projectes. Conceptes bàsics* [recurs d'aprenentatge]. Barcelona:UOC

² ESCUDERO, Xavier. *Qualitat del programari: gestió de la qualitat i mètriques* [recurs d'aprenentatge]. Barcelona:UOC

La present memòria segueix l'estructura d'aquestes fases. En primer lloc es presenten una sèrie de decisions de disseny fetes *a priori* que resulten clau per entendre el projecte. A continuació s'aprofundeix en l'establiment dels requisits i s'estableixen una sèrie de proves que conduiran la fase de disseny, seguint el paradigma del desenvolupament guiat per proves. El següent apartat fa referència a la implementació i les proves i es finalitza amb unes conclusions i una explicació sobre com podria evolucionar el projecte en el futur.

3. Principals opcions de disseny

En primer lloc, cal dir que l'ús d'un ERP diferent d'Odoo no es contempla. Es tracta d'un projecte real en què el client està acostumat a aquest ERP i li resulta útil. De fet, aquesta és només una part d'un projecte global que suposa la migració de la versió 11 a la versió 13 del *software*, un procés en què es mantenen algunes funcionalitats antigues, però que s'aprofita per a incloure algunes noves com les tractades en aquest TFG.

Pel que fa a la definició de les **dimensions** dels productes genèrics, s'ha pensat en dos alternatives:

- Aprofitar el tàndem plantilla-variant propi d'Odoo. A Odoo es poden definir plantilles de productes i, a partir d'elles, una sèrie de variants. Per a la definició de les variants, primer cal establir uns atributs a la plantilla que, en aquest cas, serien llarg, ample i gruix. A continuació s'indicarien els possibles valors que pot adquirir cadascú d'aquests atributs. En aquest context s'haurien d'indicar les diferents mides que podria adquirir cadascuna de les dimensions. Una vegada fet açò el sistema, de manera automàtica, genera una variant per a cadascuna de les possibles combinacions dels valors dels atributs.
- Indicar les dimensions del producte genèric quan se selecciona en una venda/albarà/factura. És a dir, si es vol crear una factura amb el producte 'Biga', se selecciona el producte genèric 'Biga' (sense dimensions definides) i es defineixen les dimensions d'aquesta biga en concret en la línia de factura.

La primera d'aquestes alternatives s'abandona, entre altres raons, perquè es crearien moltes variants que mai no s'utilitzarien. Tampoc sembla una opció molt versàtil, ja que s'estaria limitant el sistema a un conjunt de possibles valors definits prèviament.

Pel que fa al **càlcul del preu** dels productes genèrics, aquest s'ha de realitzar a partir de la unitat secundària del producte genèric (metres cúbics, metres quadrats o metres lineals) i un preu per unitat secundària. S'observen dues alternatives:

- Establir un preu fix per unitat secundària segons el producte.
- Generar matrius que relacionen un producte genèric que té unes determinades dimensions amb un preu per unitat secundària.

D'aquestes dues alternatives, se selecciona la segona. La raó principal és que, encara que d'alguna manera es pot establir una relació entre el volum de fusta emprat i el preu (a major volum, preu més elevat) aquesta relació no és necessàriament lineal i, per tant, caldria establir moltes casuístiques, no només depenents del producte.

Per últim, s'ha parlat de l'aplicació de **tractaments** als productes. Aquests tractaments es podrien gestionar de dues maneres diferents:

- A partir d'instàncies de productes. És a dir, es crearien productes que actuarien com a tractaments.
- A través d'una nova classe que representara els tractaments.

D'aquestes dues opcions, s'opta per la primera. En primer lloc, utilitzar instàncies del model de producte d'Odoo ajuda a estalviar temps de desenvolupament d'un nou model i ofereix ja tant uns atributs com unes operacions que es poden fer servir. En segon lloc, els tractaments es conceben, en el fons, com a serveis, i els serveis són un tipus de producte ja contemplat a Odoo. Utilitzar-los permet, per exemple, introduir-los com a línies de venda o factura i, en conseqüència, aprofitar la gestió de preus ja establida.

4. Requisites

4.1. Introducció

La següent especificació de requisits s'ha realitzat seguint les recomanacions i estructura detallada a l'*IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications* de la Computer Society.

4.1.1. Propòsit

El propòsit d'aquest document és fer una anàlisi completa per establir les funcionalitats d'un projecte de programari centrat a adaptar l'ERP Odoo a les necessitats concretes d'una empresa fustera per a, posteriorment, donar una relació detallada dels requisits, tant funcionals com no funcionals, que el producte final haurà de complir. S'inclou, així mateix, una descripció del producte mínim lliurable.

4.1.2. Abast

D'aquest desenvolupament sorgiran una sèrie de mòduls personalitzats per a l'ERP Odoo. De manera global, hauran de permetre calcular la fusta emprada amb la venda d'alguns productes, distingir entre les vendes amb fusta ecològica de les que no ho són i considerar l'aplicació de tractaments als productes en venda. Només es contempla la part de *back end*, és a dir, la utilitzada per a la gestió de l'empresa, i no es considera la part del *front end*, que es correspondria amb la pàgina web i la botiga online i que es deixaran per a futurs desenvolupaments. Per les característiques del client, tampoc farà falta tindre en consideració, almenys de moment, més rols que el d'administrador de sistema, que tindrà permisos complets, i no es consideraran usuaris amb permisos més restringits o amb requisits específics.

4.1.3. Acrònims i abreviatures

Al llarg d'aquesta memòria s'utilitzaran els següents acrònims i abreviatures

Expressió	Terme	Definició
PG	Producte Genèric	Producte per al qual es poden definir les seues dimensions, valors que serviran per al càlcul del seu preu, de la fusta emprada en la seua construcció i el seu volum, superfície o longitud segons el producte en qüestió.
PVM	Producte amb Volum Manual	Producte la fusta emprada del qual s'ha d'introduir manualment cada vegada que s'utilitza en una venda, albarà o factura.
PVC	Producte amb Volum per Coeficient	Producte en el qual el càlcul del volum de fusta emprada es realitza a través d'un coeficient.
m3	Metres cúbics	Utilitzat tant com a unitat de mesura com per especificar el tipus concret d'un PG.
m2	Metres quadrats	Utilitzat tant com a unitat de mesura com per especificar el tipus concret d'un PG.
ml	Metres lineals	Utilitzat tant com a unitat de mesura com per especificar el tipus concret d'un PG.
UoM	Unitat de Mesura	Indica en quina unitat es mesura la quantitat d'un producte. Alguns exemples són unitats, metres o quilos.
FE	Fusta emprada	Fa referència al volum de fusta que representen els productes.

Així mateix, s'utilitzen indistintament les paraules 'alt' i 'gruix' per a fer referència a una de les dimensions.

4.1.4. Resum

El present apartat d'especificació de requisits s'inicia amb una descripció general del projecte, tant del seu marc (l'ERP Odoo en aquest cas) com de les seues funcionalitats, restriccions o dependències. Una vegada presentada tota aquesta informació, es passa a la definició dels requisits en si mateixos, tant dels funcionals com dels no funcionals. Els primers apareixen dividits segons les diferents característiques del sistema que es desenvolupa.

4.2. Descripció general

4.2.1. Perspectiva del producte

El producte desenvolupat ha de ser capaç d'adaptar l'ERP Odoo a una sèrie de necessitats que una empresa fustera té en el seu treball del dia a dia, aspectes que a escala general ja s'han discutit en apartats anteriors d'aquesta memòria i que es detallaran posteriorment a través de la seua descripció i traducció a requisits.

És important tindre en compte que adaptar un sistema existent afegint-li noves funcionalitats no pot anar en contra de la manera de funcionar nativa del *software*. És a dir: ampliar-lo no hauria de provocar que altres parts del sistema deixaren de funcionar correctament. Per això és important conèixer quins són els patrons existents relacionats amb les noves aplicacions per veure com cal adaptar-los.

Les exigències del client es relacionen amb una sèrie d'entitats presents a Odoo, que es descriuen a continuació. Les funcionalitats introduïdes amb el desenvolupament hauran de mantindre la base d'Odoo, que s'haurà de compatibilitzar amb les novetats.

4.2.1.1. Productes

Per a la definició de productes a Odoo s'utilitza la jerarquia formada per les classes ProductTemplate, o plantilla de producte, i ProductProduct, o variant de producte (Diagrama 1).

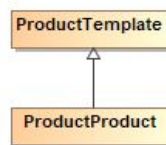
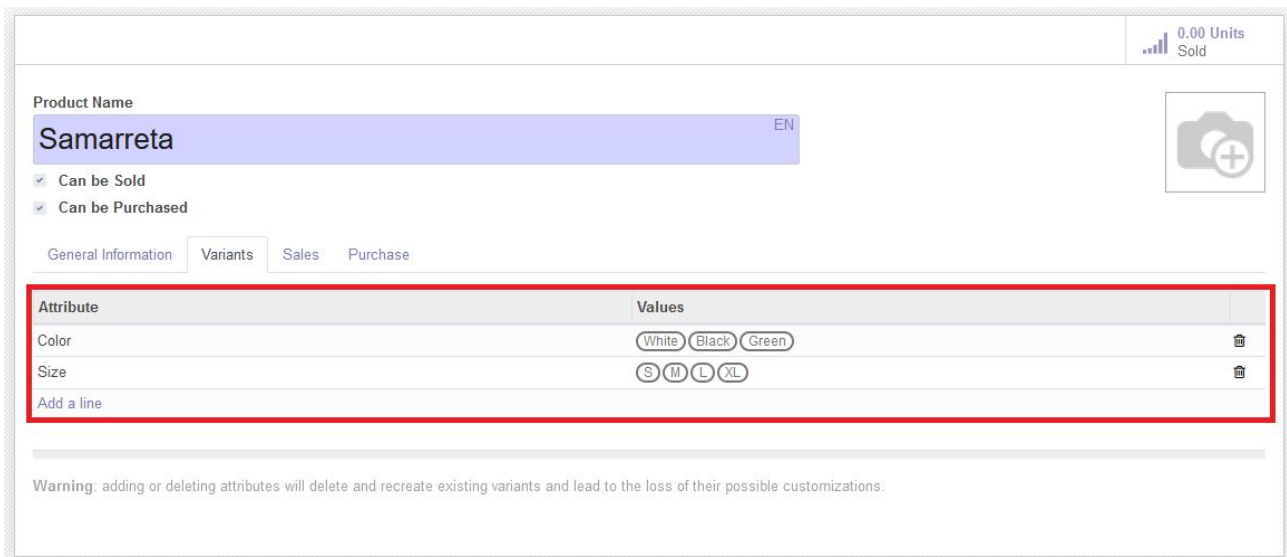


Diagrama 1

La variant de producte heretarà les característiques de la plantilla de producte, encara que hi podrà modificar alguna d'elles i afegir-ne de noves. Les variants de producte són el resultat d'introduir atributs a la plantilla de producte amb diferents valors possibles. De la combinació de tots aquests atributs i els seus valors possibles sorgiran, de manera automàtica, totes les variants (aquest és el funcionament per defecte, encara que seria possible configurar-lo perquè es generaren les variants a mesura que es necessiten).

Un exemple il·lustratiu podria ser el producte 'Samarreta'. Es crea una nova plantilla de producte amb aquest nom i s'estableixen dos atributs: talla i color. Els valors possibles de l'atribut talla serien 'S', 'M', 'L' i 'XL'. Els valors possibles de l'atribut color serien 'blanc', 'negre' i 'verd', tal com es veu a la Captura 1.



Captura 1

El resultat de realitzar aquesta configuració d'una plantilla de producte seria obtenir 12 variants després d'aplicar totes les combinacions possibles dels valors dels atributs, tal com es veu a la Captura 2.

Internal Reference	Name	Attribute Values	Sales Price	Cost Price
<input type="checkbox"/> Sam-01	Samarreta	Color: White Size: S	20.00	5.00
<input type="checkbox"/> Sam-02	Samarreta	Color: White Size: M	20.00	5.00
<input type="checkbox"/> Sam-03	Samarreta	Color: White Size: L	20.00	5.00
<input type="checkbox"/> Sam-04	Samarreta	Color: White Size: XL	20.00	5.00
<input type="checkbox"/> Sam-05	Samarreta	Color: Black Size: S	20.00	5.00
<input type="checkbox"/> Sam-06	Samarreta	Color: Black Size: M	20.00	5.00
<input type="checkbox"/> Sam-07	Samarreta	Color: Black Size: L	20.00	8.00
<input type="checkbox"/> Sam-08	Samarreta	Color: Black Size: XL	20.00	8.00
<input type="checkbox"/> Sam-09	Samarreta	Color: Green Size: S	20.00	8.00
<input type="checkbox"/> Sam-10	Samarreta	Color: Green Size: M	20.00	8.00
<input type="checkbox"/> Sam-11	Samarreta	Color: Green Size: L	20.00	8.00
<input type="checkbox"/> Sam-12	Samarreta	Color: Green Size: XL	20.00	8.00

Captura 2

Alguns dels atributs de les variants de producte, com per exemple el preu, s'hereten de la plantilla. D'altres, com el cost, són propis de cadascuna de les variants.

Una particularitat d'Odoo és que generalment no es treballa amb les plantilles de producte, sinó amb les variants de producte. Tant és així que, en el moment en què es crea una nova plantilla de producte es genera, automàticament, una variant sense atributs que hereta per complet les característiques de la *template*.

Un altre aspecte important relacionat amb els productes és la unitat de mesura (UoM). Aquesta indica amb quines unitats es compren i venen per defecte els productes. Algunes de les unitats preconfigurades són 'Unitats', 'Hores', 'Metres' o 'Quilograms'. A una UoM se li pot aplicar una categoria (quantitat, temps, longitud, pes...). D'aquesta manera, totes les UoM d'una mateixa categoria poden convertir-se en una altra de la mateixa categoria gràcies a un índex de conversió.

Continuant amb l'exemple del producte 'samarreta', el qual porta com a UoM la unitat 'Unitats', si es ven la quantitat de 6 unitats de samarretes, a 20 euros cadascuna, el preu total serà de 120 euros. Però també es podrien vendre 6 dotzenes de samarretes a 20 euros cadascuna de les samarretes (ja que 'Units' i 'Dozens' són de la mateixa categoria) i gràcies a l'índex de conversió entre unitats de la mateixa categoria es calcularia, d'una banda, que el preu total és de 1440 euros i, d'altra, que l'estoc s'ha de reduir en 72 unitats de samarretes.

La UoM es defineix a la plantilla de producte i s'hereta per totes les variants, la qual cosa vol dir que totes les variants d'una mateixa *template* tindran la mateixa UoM.

4.2.1.2. Vendes

A Odoo, una venda és una instància que conté quins són els productes que un client determinat adquireix. Una venda passa per dos estats principals: pressupost (quan la venda encara no ha estat confirmada pel client) i ordre de venda (quan la venda ja ha estat confirmada). Els elements principals que conté una venda són el client i les línies de producte, on s'indiquen els béns que el comprador vol adquirir juntament amb la quantitat, el preu o els impostos (Captura 3).

The screenshot shows the Odoo interface for a quotation (S00021) in the 'Sales Order' state. The interface includes a navigation bar with 'Estat de la venda' and 'Accés a albarà' buttons. The main content area is divided into sections for customer information and product lines.

Dades del client

Customer	Deco Addict 77 Santa Barbara Rd Pleasant Hill CA 94523 United States	Order Date	10/14/2020 18:43:41
		Payment Terms	30 Days

Línies de productes

Product	Description	Quantity	Delivered	Invoiced	Unit Price	Tax...	Subtotal	
[DESK0004] Customizable Desk (CONFIG...)	[DESK0004] Customizable Desk (CONFIG) (Aluminium, Black) 160x80cm, with large legs.	2.000	0.000	0.000	800.40		\$ 1,600.80	
[E-COM06] Corner Desk Right Sit	[E-COM06] Corner Desk Right Sit	8.000	0.000	0.000	147.00		\$ 1,176.00	

Summary:

- Untaxed Amount: \$ 2,776.80
- Taxes: \$ 0.00
- Total: \$ 2,776.80

Captura 3

En el moment en què la venda passa a l'estat 'Ordre de venda' (*Sales Order*) es genera un albarà, que no és més que un moviment d'estoc. La informació rellevant per a l'albarà, com els productes adquirits amb la seua quantitat, es transfereix automàticament de la venda a l'albarà. Així mateix, una vegada s'ha validat la venda també es pot generar automàticament una factura amb les dades introduïdes. Cal tindre en compte que només es traspassaran a l'albarà aquells productes que siguin emmagatzemables o consumibles (els serveis no es traspassen, ja que no es poden "transportar").

4.2.1.3. Albarans

Un albarà representa un moviment d'estoc. Els albarans poden ser tant d'entrada (quan l'empresa adquireix béns) o d'eixida (quan es venen productes). En el desenvolupament tractat en aquest projecte, interessa la tipologia d'albarans d'eixida.

Si l'albarà s'ha generat a partir d'una venda, es traspassen, entre d'altres, la informació del client, els productes afegits a les línies de venda així com la seua quantitat (els albarans no contenen informació de preus). Validar un albarà significa traure productes del magatzem de l'empresa, és a dir, reduir les existències. En aquest context, es poden trobar dues situacions:

- **Es disposa de prou unitats.** L'estoc de l'empresa té tota la quantitat de producte demandada pel client. En aquest cas, quan es valide l'albarà, es restarà de l'estoc la quantitat de producte comprada (Captura 4).

Product	Demand	Reserved	Done	Unit of Measure
[DESK0004] Customizable Desk (CONFIG) (Aluminium, Black)	2.000	2.000	0.000	Units
[E-COM06] Corner Desk Right Sit	8.000	8.000	0.000	Units

Captura 4

Tal com es veu a la Captura 4, a la pestanya 'Operacions' de l'albarà podem veure que la quantitat reservada de producte coincideix amb la demandada. Això és així perquè n'hi ha prou unitats a l'estoc.

Com es mostra a la Captura 5, des de la pestanya 'Operacions detallades' es pot seleccionar quantes de les unitats reservades s'entreguen al client. Que la quantitat realitzada ('Done') coincidisca amb la demandada vol dir que l'albarà s'ha entregat per complet.

Product	Lot/Serial Number	Reserved	Done	Unit of Measure
[DESK0004] Customizable Desk (CONFIG) (Aluminium, Black)		2.000	2.000	Units
[E-COM06] Corner Desk Right Sit		8.000	8.000	Units

Captura 5

- **No es disposa de prou unitats.** En aquest cas, no es podrà servir l'albarà complet i el sistema donarà l'opció de crear una *backorder* o albarà parcial, és a dir, un nou albarà amb la quantitat pendent. Aquest nou albarà funcionarà de la mateixa manera que qualsevol altra entitat d'aquest tipus. És a dir, de nou s'haurà de comprovar si les existències a l'estoc permeten servir tota la quantitat demandada pel client. Si no es pot, es podria generar una nova *backorder* i així successivament.

4.2.1.4. Factures

Com les vendes, les factures contenen una llista amb els productes que un client adquireix. Conceptualment, la principal diferència és que una factura oficialitza la transacció. Les factures es poden trobar en dos estats: esborrany i publicada. Mentre que en l'estat esborrany l'usuari pot realitzar modificacions en les dades de la factura, una vegada aquesta ha estat publicada, ja no es permet realitzar modificacions. Com amb les vendes, les factures contenen la informació de client i un conjunt de línies amb informació dels productes adquirits (Captura 6).

Customer Invoice
INV/2020/0003

Dades del client

Customer: Deco Addict
77 Santa Barbara Rd
Pleasant Hill CA 94523
United States

Invoice Date: 10/08/2020
Payment Terms: 10/08/2020

Reference

Invoice Lines

Product	Label	Quantity	Price	Taxes	Subtotal
[FURN_8999] Three-Seat Sofa	[FURN_8999] Three-Seat Sofa Three Seater Sofa with Lounger in Steel Grey Colour	5.000	90.00		\$ 450.00
[FURN_8220] Four Person Desk	[FURN_8220] Four Person Desk Four person modern office workstation	5.000	15.00		\$ 75.00

Línies de producte

Untaxed Amount: \$ 525.00
Total: \$ 525.00
Amount Due: \$ 525.00

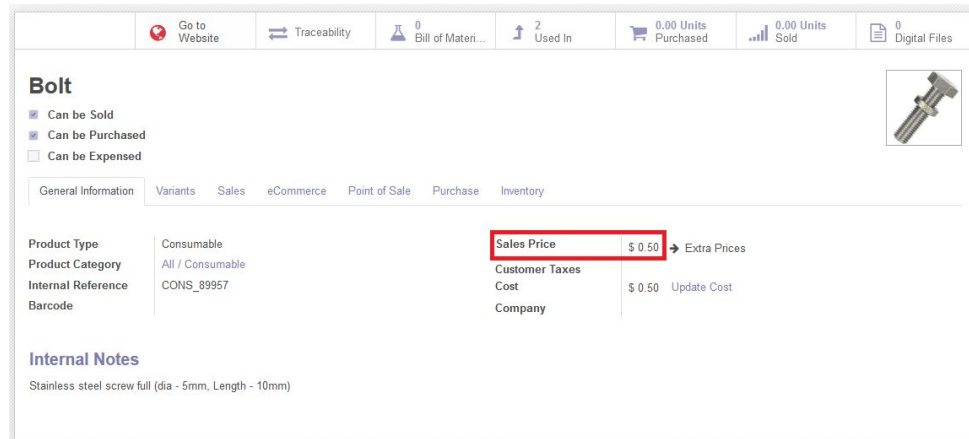
Estat de la factura: Draft, Posted

Captura 6

Una factura es pot crear aïlladament, seleccionant la informació i els productes de la mateixa manera que es faria quan es crea una venda. Així i tot, la manera habitual de procedir del client és generar la factura a partir de la venda. De la mateixa manera que ocorre amb els albarans, la informació de la venda es traspassa a la factura.

4.2.1.5. Preus i *Pricelists*

Un dels elements configurables dels productes és el seu preu, de manera que, quan se selecciona un producte a una venda o factura, el sistema mostra automàticament quant costa (Captura 7).



Captura 7

Però a Odoo també es poden establir tarifes o *pricelists*, les quals permeten determinar preus diferents per a un mateix producte. Aquest concepte és interessant en tant que els negocis moltes vegades no apliquen els mateixos preus a tots els seus clients, o les quanties varien segons la quantitat adquirida.

La selecció d'una *pricelist* es pot realitzar manualment des de la venda o la factura, encara que també és possible vincular un contacte amb una *pricelist* determinada. D'aquesta manera, quan se seleccione aquest contacte com a client, el sistema actualitzarà automàticament el camp *pricelist* i, en conseqüència, els preus que es mostren quan se seleccione un producte. Entre altres coses, amb una *pricelist* és possible establir tant un preu determinat per a un producte com fixar descomptes, i l'aplicació d'una estratègia o una altra es pot realitzar amb independència de la quantitat adquirida o a partir d'una quantitat determinada.

Existeix una tarifa per defecte a Odoo anomenada 'Public Pricelist', la qual agrupa els preus establits des de les fitxes de producte. Aquesta serà la tarifa que s'aplique quan el contacte client no tinga una *pricelist* adjudicada i no s'especifique que es vol aplicar una tarifa diferent.

4.2.2. Funcionalitats del producte

De manera general, el producte desenvolupat ha de tindre les següents funcionalitats, que han d'operar correctament dins del sistema general Odoo:

- **Tipologia de productes**
- **Càlcul de la fusta emprada i del volum, superfície o longitud dels PG**
- **Càlcul dels preus dels PG**
- **Vendes factures i albarans amb fusta PEFC**
- **Aplicació de tractaments**
- **Generació de documents amb nova informació**

A continuació s'aprofundeix en cadascuna d'aquestes funcionalitats.

4.2.2.1. Tipologia de productes

El projecte requereix la definició d'una classificació dels productes. Els productes del sistema poden ser d'algun d'aquests tipus o poden no tindre cap tipus especial i comportar-se de la manera habitual d'Odoo. La tipologia de productes serveix de base per a moltes de les funcionalitats del desenvolupament que s'expliquen posteriorment.

- **Productes genèrics (PG).** Són productes customitzables segons les seues dimensions i és per això que serà necessari introduir-les quan s'afegeixen a les vendes, factures i albarans. Per exigències del client, el llarg s'ha d'establir en metres, mentre que l'ample i el gruix s'indiquen en centímetres. Els PG poden dividir-se en quatre subtipus:
 - **PG en metres cúbics (m3).** Per exemple, una biga.
 - **PG en metres quadrats (m2).** Per exemple, un tauler.
 - **PG en metres lineals (ml).** Per exemple, un llistó.
 - **PG de tipus pal.**
- **Productes amb introducció manual del volum (PVM).** El client necessita disposar d'un tipus de producte per als quals se'ls puga introduir manualment el valor de FE quan s'afegeixen a vendes, albarans i factures. Un exemple d'aquest tipus de producte és "camió de troncs". Aquest mateix producte podria suposar un volum de fusta X en una ocasió però un volum de fusta Y en altra. És a dir, no deixa de ser el mateix producte, però amb una especificació diferent pel que fa a la fusta que representa.
- **Productes amb un coeficient de càlcul de la FE a partir de la seua quantitat (PVC).** Aquests productes han de tindre configurat un multiplicador que, aplicat a la quantitat de producte, calcula la FE.

4.2.2.2. Càlcul de fusta emprada i volum, superfície o longitud dels PG

Cada venda, albarà i factura haurà de tindre un registre de la FE. Els càlculs necessaris dependran de la tipologia del producte. També a les diferents línies de les tres entitats s'hauran de calcular els m3, m2 o ml dels PG segons el seu subtipus, tenint en compte que els de subtipus pal fan servir metres lineals i els PVM i PVC utilitzen els metres cúbics com a unitat secundària.

- **Productes genèrics (PG).**

- **PG en metres cúbics (m3).** El càlcul dels m3 i de la FE (total de la línia) es realitza amb la fórmula següent:

$$m3 = llarg \cdot (gruix/100) \cdot (ample/100) \cdot quantitat \cdot factor_conversió_UoM$$
$$FE = m3$$

- **PG en metres quadrats (m2).** El càlcul dels m2 i de la FE (total de la línia) es realitza amb la fórmula següent:

$$m2 = llarg \cdot (ample/100) \cdot quantitat \cdot factor_conversió_UoM$$
$$FE = m2 \cdot (gruix/100)$$

- **PG en metres lineals (ml).** Els càlculs dels ml i de la FE (total de la línia) són els següents:

$$ml = llarg \cdot quantitat \cdot factor_conversió_UoM$$
$$FE = ml \cdot ((ample/100) \cdot (gruix/100))$$

- **PG de tipus pal.** Els càlculs i de la FE es realitzen de la següent manera:

$$ml = llarg \cdot quantitat \cdot factor_conversió_UoM$$
$$FE = \pi \cdot (((gruix/100)+(ample/100))/4)^2 \cdot ml$$

- **Productes amb introducció manual del volum (PVM).** Quan s'utilitzen aquests productes, a més d'enregistrar la FE (dada que cal introduir manualment) també cal considerar els m3, de la mateixa manera que es feia amb els PG de tipus m3. Així:

$$m3 = FE$$

- **Productes amb càlcul de la fusta emprada a través d'un coeficient (PVC).** Els m3 i la FE (valor total de la línia) es calculen a través del coeficient de la següent manera:

$$FE = coeficient \cdot quantitat \cdot factor_conversió_UoM$$
$$M3 = FE$$

Exemple pràctic

Considerem que en una venda es volen introduir els següents productes:

- 3 unitats de taulers de 2 metres de llarg, 60 cm d'ample i 3 cm de gruix. Es tracta d'un producte PG amb m2 com a unitat secundària. Per tant:
 - $m2 = llarg \cdot (ample/100) \cdot quantitat \cdot factor_conversió_UoM = 3.6 m2$
 - $FE = m2 \cdot (gruix/100) = 0.108m3$
- 2 dotzenes de pals d'1,5 metres de llarg, 2 cm de gruix i 3 cm d'ample. Es tracta d'un PG de tipus pal. Per tant:
 - $ml = llarg \cdot quantitat \cdot factor_conversió_UoM = 36 m$
 - $FE = \pi \cdot (((gruix/100)+(ample/100))/4)^2 \cdot ml = 0,018 m3$
- 1 camió de troncs amb 24 m3 de fusta. Es tracta d'un producte PVM i, per tant, s'introdueix el volum total de fusta manualment.
 - $m3 = 24 m3$
 - $FE = 24 m3$
- 30 kg de serradures. Es tracta d'un producte PVC amb un índex de conversió d'1,2.
 - $m3 = 36 m3$
 - $FE = 36 m3$

En resum, les línies de venda apareixerien de la següent manera:

Producte	Quantitat	UoM	Llarg	Ample	Gruix	m3	m2	ml	FE
Tauler	3	Unitat	2	60	3		3.6		0.108
Pal	2	Dontzena	1,5	3	2			36	0,018
Camió troncs	1	Unitat				24			24
Serradures	30	kg				36			36

La venda registraria que el total de fusta emprada és 60.126 m3.

4.2.2.3. Càlcul dels preus

A conseqüència de l'establiment a l'apartat anterior d'alguns tipus de productes amb particularitats, el càlcul dels preus a les vendes i factures s'ha d'adaptar a aquestes necessitats.

Els preus dels PG dependran de les dimensions que s'indiquen en cada cas, i caldrà mostrar aquest preu quan se seleccionen aquests productes a les vendes i factures. La definició d'aquests preus es realitza mitjançant una sèrie de matrius de preus que relacionen les dimensions dels PG amb un preu determinat per unitat secundària. Una mateixa matriu es pot utilitzar per a diversos productes i es pot vincular a diverses *pricelists*.

Serà necessari també poder modificar el preu de qualsevol producte manualment, una opció que ja contempla Odoo i que s'aprofitarà per a establir els preus dels productes PVM, ja que, en aquests tipus de productes, la fusta emprada s'introdueix manualment cada vegada que s'utilitzen i no es contempla cap sistema de càlcul automàtic de preus.

Els preus dels PVC funcionen de la manera habitual d'Odoo, definida a l'apartat 4.2.1.5.

En resum, els preus dels productes segons el seu tipus s'estableix de la següent manera:

	Càlcul de preu normal	Matriu de preus	Introducció manual
Producte normal	x		x
PG		x	x
PVM			x
PVC	x		x

4.2.2.4. Vendes, factures i albarans amb fusta PEFC

Part dels productes que el client ven porta la certificació PEFC, que indica l'explotació sostenible dels recursos forestals. Així, el client desitja indicar a les vendes, albarans i factures si la fusta emprada porta aquesta certificació o no. L'objectiu és doble:

- D'una banda, és necessari saber quins productes s'han de servir al client. Per exemple, la biga que cal lliurar no serà la mateixa si porta o no porta aquest segell.
- D'altra, es vol portar un seguiment de les comandes realitzades amb fusta certificada amb PEFC, una dada que servirà internament per a tindre un control de la producció.

No es contempla que dins d'una mateixa venda, albarà o factura alguns productes porten certificat PEFC i altres no el porten.

4.2.2.5. Aplicació de tractaments

Alguns dels productes venuts pel client accepten tractaments com, per exemple, pintura, poliment o vernís. Quan es trie algun d'aquests productes en una venda caldrà indicar si se li aplica algun d'aquests tractaments. Tindre constància d'açò respon a dues raons:

- Incrementar el preu del producte, ja que el preu d'una biga no podrà ser el mateix si està, per exemple, pintada o si no ho està.
- Saber internament quin és el producte exacte que el comprador necessita.

El preu dels tractaments s'ha d'establir sempre de manera manual. És a dir, que el càlcul del preu dels tractaments no cal que es realitzi automàticament.

S'han de fer tres puntualitzacions:

- No només els productes especials (definitos a l'apartat anterior) podrien acceptar tractament. També els productes definits d'una manera "normal" a Odoo els podrien acceptar.
- No és necessari que un producte que pugui acceptar tractaments pugui acceptar tots els tractaments disponibles. És a dir, es podrien tindre dos productes X i Y que accepten tractaments, però que el producte X accepti pintura i vernís i el producte Y només poliment.
- L'aplicació dels tractaments es realitzarà només des de les vendes i després aquesta informació es trasllada als albarans i les factures.

4.2.2.6. Generació de documents amb nova informació

A més de la introducció i gestió en el sistema de totes les dades esmentades als apartats anteriors, es necessita que aquesta informació siga accessible als documents PDF de vendes, albarans i factures generats a través d'Odoo. Açò no és important només internament, sinó també per als compradors, perquè tinguen constància exactament de quins productes adquireixen i de les seues característiques.

4.2.2.7. Dependències entre funcionalitats

Al Diagrama 2 es mostren les dependències existents entre les diferents funcionalitats.

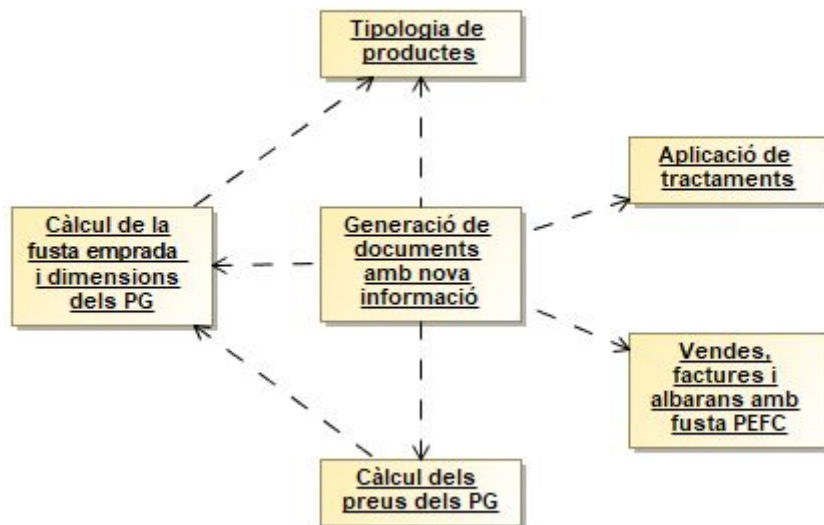


Diagrama 2

- “Càlcul de la fusta emprada i dimensions dels PG” depèn de “Tipologia de productes”, ja que les operacions de càlcul dependran tant de la tipologia del producte seleccionat com d’altres aspectes com les dimensions especificades.
- “Càlcul dels preus dels PG” depèn de “Càlcul de la fusta emprada i dimensions dels PG”, ja que el preu d’un PG es determina segons els metres cúbics, metres quadrats o metres lineals dels PG.
- “Generació de documents amb nova informació” depèn de tota la resta de funcionalitats, ja que la informació que es mostrarà procedirà de la incorporada amb totes.
- “Tipologia de productes” no té dependències cap a altres funcionalitats, ja que és, precisament, la que funciona de base per a altres funcionalitats.
- “Vendes, factures i albarans amb fusta PEFC” no té dependències cap a altres funcionalitats, ja que aquesta funcionalitat consisteix, simplement, a indicar si una d’aquestes tres entitats utilitza o no fusta PEFC.
- “Aplicació de tractaments” no té dependències cap a altres funcionalitats perquè els tractaments no tenen en consideració que els productes siguin d’alguna de les tipologies introduïdes.

4.2.3. Característiques de l'usuari

En principi, aquest *software* només serà utilitzat pel director de l'empresa client (a partir d'ara 'gestor'). Les seues característiques principals, que poden tindre un impacte en el desenvolupament del producte, són les següents:

- **Usuari habitual d'Odoo.** El gestor ja utilitza l'ERP Odoo i coneix, per tant, com funciona així com moltes de les seues possibilitats. El fet que conega el *software* fa que no només pugui expressar què és el que vol sinó també, de vegades, com ho vol.
- **Director de l'empresa.** Que l'usuari del *software* siga, al mateix temps, el director de l'empresa permet que tinga una visió global de tot el que s'espera del desenvolupament. Un avantatge és que la comunicació és directa amb la persona que fa, alhora, dels dos *stakeholders*: usuari i director de la companyia.

4.2.4. Restriccions de desenvolupament

El present treball consisteix a ampliar les funcionalitats base de l'ERP Odoo. Això vol dir que, en trobar-se dins d'un marc determinat, existeixen una sèrie de qüestions de les quals el *software* ja s'encarrega; al mateix temps, però, s'estableixen una sèrie de restriccions al desenvolupament que s'hauran de considerar perquè una nova peça encaixe a la perfecció dins d'un sistema molt complex. En definitiva, al mateix temps que no cal preocupar-se per molts dels aspectes, ja que són gestionats nativament pel *software*, açò mateix estableix una sèrie de restriccions o normes que s'han de seguir per tal que el conjunt siga totalment funcional i que es pugui ampliar en un futur si fos necessari.

4.2.4.1. Restriccions funcionals

El fet que el desenvolupament forme part d'un sistema més ampli estableix una sèrie de restriccions en l'àmbit funcional relacionades amb el fet que les noves funcionalitats no poden produir un funcionament incorrecte de la resta de peces del programari. En aquest sentit, es pot parlar de diferents restriccions:

- Odoo disposa d'un **flux** de treball bàsic del qual s'ha parlat anteriorment (Vendes → Albarans → Factures). L'adaptació del *software* a les necessitats específiques del client no pot anar en contra d'aquest flux. En primer lloc, no es tracta d'una sèrie de passos aïllats, sinó que existeixen dades que es transfereixen d'una fase a una altra i això ha de continuar realitzant-se. Algunes de les restriccions en aquest sentit són les següents:

- Els productes adquirits, les seues quantitats, els seus preus i altres **característiques passen de la venda als albarans i les factures**. Els productes especials dissenyats per al client també hauran de funcionar d'aquesta manera. Sembla, per tant, lògic considerar que aspectes com la fusta emprada o les dimensions dels PG hauran de seguir aquest flux, al mateix temps que el flux original es manté. Per tal d'obtenir aquesta transferència de dades, Odoo disposa d'una sèrie de funcions ja definides que determinen de quina manera es traspassen les dades d'un model a un altre, funcions que depenen tant del model d'origen com del model de destinació.
- Anteriorment, quan s'ha parlat dels albarans, s'ha comentat la idea dels **albarans parcials**. És a dir, que a partir d'un albarà en què s'han de servir, per exemple, 10 productes, en cas que es distribuïsquen menys unitats, es pot crear un segon albarà amb la diferència en la quantitat. Açò crearia una nova restricció: quan es creen albarans parcials, la suma de la fusta emprada en tots ells haurà de ser igual a la fusta establida a la venda de la qual procedeixen.
- A Odoo, les línies de vendes, albarans i factures es relacionen amb instàncies de **variants de producte**, no de plantilles de productes. Aquest desenvolupament haurà de tindre açò en consideració, ja que no serà possible fer servir les plantilles de productes en sí mateixes, sinó que s'hauran d'utilitzar les variants.
- La **UoM** és un dels elements que es defineix a la plantilla i hereten totes les variants. En la definició dels nous tipus de productes, concretament dels PG, s'ha parlat de productes en m3, m2, ml i pal. El fet que les UoM es mantinguen en totes les variants d'una plantilla determinada fa que la unitat de mesura de la fusta, o unitat secundària, haja d'estar també definida a la plantilla.
- Les vendes i factures porten associada una i només una **tarifa**. Açò s'haurà d'aplicar també als productes PG, quan s'hagen de definir les matrius que relacionen les dimensions amb un preu determinat.

4.2.4.2. Restriccions del llenguatge

Un altre aspecte molt important pel que fa al desenvolupament són els diferents llenguatges emprats. Odoo segueix un patró Model-Vista-Controlador, i cadascun d'aquests components està desenvolupat fent servir llenguatges diferents: mentre que els models i els controladors es desenvolupen fent servir el llenguatge **Python**, les vistes es fan en **xml**.

L'ERP disposa d'una manera específica de definir noves classes o realitzar herències que s'haurà de respectar, independentment que la creació d'una nova classe a Python fora d'aquest *framework* es pugui realitzar de manera diferent. Odoo també disposa d'una API amb què decorar funcions i d'una manera específica de definir el tipus dels atributs d'una classe. Tots aquests aspectes estan aprofundits a la secció de disseny.

4.2.4.3. Restriccions d'interfície gràfica

Aquest projecte es limita al *back end* d'Odoo, ja que el desenvolupament fa referència a una sèrie d'eines addicionals perquè en dispose el gestor. Així, la part web o de *front end* queda fora de l'abast. Açò té un impacte sobre les restriccions, ja que el *back end* d'Odoo n'aporta moltes pel que fa a les interfícies gràfiques. Açò no vol dir que siga impossible transformar radicalment les vistes o inclús reemplaçades per altres noves, però l'arquitectura del *software* no està dissenyada per recolzar grans modificacions visuals del *back end* (encara que sí ho està per tindre una part web o de *front end* molt personalitzable).

Les vistes base d'Odoo tenen una estètica uniforme. D'una banda, les modificacions de vistes existents es realitzen heretant altres vistes per a afegir o ocultar camps o aconseguir xicotets canvis de visualització (assenyalar certa informació amb altres colors, canviar les condicions en què es mostra o s'oculta un camp...). D'altra banda, es poden crear vistes totalment noves a través de fitxers xml, però la renderització mantindrà l'estil de presentació. Plantejar-se, per tant, pantalles al *back end* amb estètica diferent es pot rebutjar directament, ja que el seu cost seria massa elevat.

4.2.4.4. Restriccions de seguretat

Si bé és cert que els permisos d'accés als models base d'Odoo ja són gestionats directament pel *software* i, per tant, no cal preocupar-se per definir nous sistemes, en cas de crear nous models, la manera de gestionar els permisos d'accés està establida pel *software* i s'haurà de respectar.

4.2.5. Suposicions i dependències

Odoo 13 funciona directament sobre navegadors web i està dissenyat perquè s'utilitze Google Chrome, Mozilla Firefox, Microsoft Edge o Apple Safari³. Açò vol dir que l'ús d'altres navegadors podria provocar que el funcionament no fora l'òptim. Existeix també una app d'Odoo, però no cal adaptar les vistes a terminals mòbils.

Així mateix, el desenvolupament està pensat perquè pugua funcionar amb la versió 13 de l'ERP. En cas de decidir-se utilitzar una altra versió, ja fora una anterior o posterior, seria necessari replantejar part dels requisits. Així mateix, el *software* està compost d'un conjunt de mòduls i s'hauran d'instal·lar alguns d'ells per tindre les funcionalitats base sobre les quals es recolza aquest projecte.

³ *Deploying Odoo* [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.odoo.com/documentation/13.0/setup/deploy.html#supported-browsers>

4.3. Requisits específics

Per a l'especificació de requisits, s'ha seguit la plantilla organitzada per característiques proposada al *IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications* de la Computer Society. En primer lloc, es presenten requisits d'interfície externa, concretament les interfícies d'usuari. A continuació es presenten les diferents característiques del sistema (que es corresponen amb les funcionalitats establides a l'apartat 4.2.2.) amb els seus requisits funcionals associats. Finalment, es parla de requisits de rendiment, restriccions de disseny, atributs del sistema de *software* i d'altres requisits no funcionals.

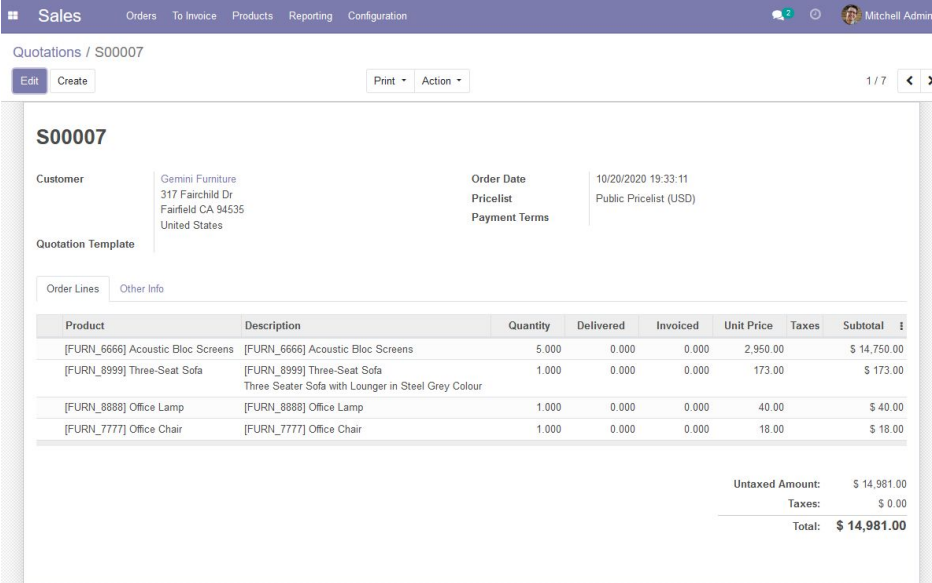
4.3.1. Requisits d'interfície externa

4.3.1.1. Interfície d'usuari

Abans de la implementació, s'estableix quina és la informació que hauria d'aparèixer en les diferents pantalles que intervenen en el sistema. En el cas de les vistes heretades, es concreten quins elements addicionals s'han d'afegir; en el cas de les vistes noves, s'especifiquen tots els elements que s'han de mostrar.

Vendes

Es tracta d'una vista heretada, l'aspecte base de la qual és el mostrat a la Captura 8.



Product	Description	Quantity	Delivered	Invoiced	Unit Price	Taxes	Subtotal
[FURN_6666] Acoustic Bloc Screens	[FURN_6666] Acoustic Bloc Screens	5.000	0.000	0.000	2.950.00		\$ 14,750.00
[FURN_8999] Three-Seat Sofa	[FURN_8999] Three-Seat Sofa Three Seater Sofa with Lounger in Steel Grey Colour	1.000	0.000	0.000	173.00		\$ 173.00
[FURN_8888] Office Lamp	[FURN_8888] Office Lamp	1.000	0.000	0.000	40.00		\$ 40.00
[FURN_7777] Office Chair	[FURN_7777] Office Chair	1.000	0.000	0.000	18.00		\$ 18.00

Untaxed Amount: \$ 14,981.00
Taxes: \$ 0.00
Total: **\$ 14,981.00**

Captura 8

Caldria afegir els següents elements:

- Tres columnes addicionals a les línies de producte per a introduir les **dimensions** dels PG. Els camps estaran desactivats si el producte no és de tipus PG.
- Tres columnes addicionals a les línies de producte on es mostren els **metres cúbics**, **metres quadrats** o **metres lineals**. Si el producte no és PG, PVM o PVC, els tres camps estaran desactivats. Si el producte és PG només estarà activat el camp del subtipus específic (m3, m2 o ml). En els PVM i PVC estarà activa la columna dels m3.
- Una columna en què es mostre el **preu per unitat secundària (m3, m2 o ml⁴)**. Aquesta casella estarà desactivada si el producte no és PG.
- Una casella on s'especifique si la venda utilitza productes amb fusta **PEFC**.
- Un camp amb la **fusta emprada** a la comanda.
- Una opció per a afegir **tractaments** als productes.

Factures

Es tracta d'una vista heretada, l'aspecte base de la qual és el mostrat a la Captura 9:

The screenshot displays a web-based invoicing system. At the top, there's a navigation bar with 'Invoicing' and various menu items like 'Customers', 'Vendors', 'Reporting', and 'Configuration'. Below this, the current view is identified as 'Invoices / INV/2020/0003'. The main content area is titled 'Customer Invoice' and shows the following details:

- Customer:** Deco Addict, 77 Santa Barbara Rd, Pleasant Hill CA 94523, United States.
- Invoice Date:** 10/08/2020
- Payment Terms:** 10/08/2020
- Company:** My Company (San Francisco)

Below the details, there's a table for 'Invoice Lines' with the following data:

Product	Label	Quantity	Price	Taxes	Subtotal
[FURN_8999] Three-Seat Sofa	[FURN_8999] Three-Seat Sofa Three Seater Sofa with Lounger in Steel Grey Colour	5.000	90.00		\$ 450.00
[FURN_8220] Four Person Desk	[FURN_8220] Four Person Desk Four person modern office workstation	5.000	15.00		\$ 75.00

At the bottom right of the invoice, the following summary is shown:

- Untaxed Amount: \$ 525.00
- Total: **\$ 525.00**
- Amount Due: **\$ 525.00**

Captura 9

⁴ Els metres lineals són la unitat secundària tant dels PG de tipus ml com dels PG de tipus pal.

Caldrà afegir la següent informació:

- Tres columnes addicionals a les línies de producte per a introduir les **dimensions** dels PG. Els camps estaran desactivats si el producte no és de tipus PG.
- Tres columnes addicionals a les línies de producte on es mostren els **metres cúbics, metres quadrats** o **metres lineals**. Si el producte no és PG, PVM o PVC, els tres camps estaran desactivats. Si el producte és PG només estarà activat el camp del subtipus específic (m3, m2 o ml). En els PVM i PVC estarà activa la columna dels m3.
- Una columna en què es mostre el **preu per unitat secundària (m3, m2 o ml)** de fusta emprada. Aquesta casella estarà desactivada si el producte no és PG.
- Una casella s'especifique si la venda utilitza productes amb fusta **PEFC**.
- Un camp amb la **fusta emprada** a la comanda.
- Un camp per indicar la *pricelist*.

Albarans

Es tracta d'una vista heretada, l'aspecte base de la qual és el de la captura 10:

Product	Demand	Done	
[E-COM07] Large Cabinet	50,000	50,000	

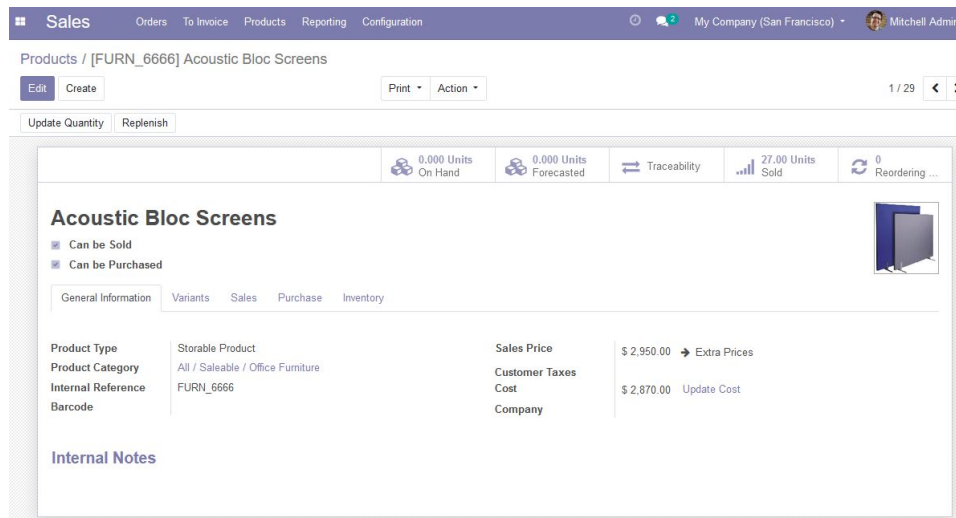
Captura 10

Caldrà afegir la següent informació:

- Tres columnes addicionals a les línies de producte per a introduir les **dimensions** dels PG. Els camps estaran desactivats si el producte no és de tipus PG.
- Tres columnes addicionals a les línies de producte on es mostren els **metres cúbics, metres quadrats** o **metres lineals**. Si el producte no és PG, PVM o PVC, els tres camps estaran desactivats. Si el producte és PG només estarà activat el camp del subtipus específic (m3, m2 o ml). En els PVM i PVC estarà activa la columna dels m3.
- Una casella on s'especifique si la venda utilitza productes amb fusta **PEFC**.

Plantilla de producte

Es tracta d'una vista heretada que té l'aspecte de la Captura 11:



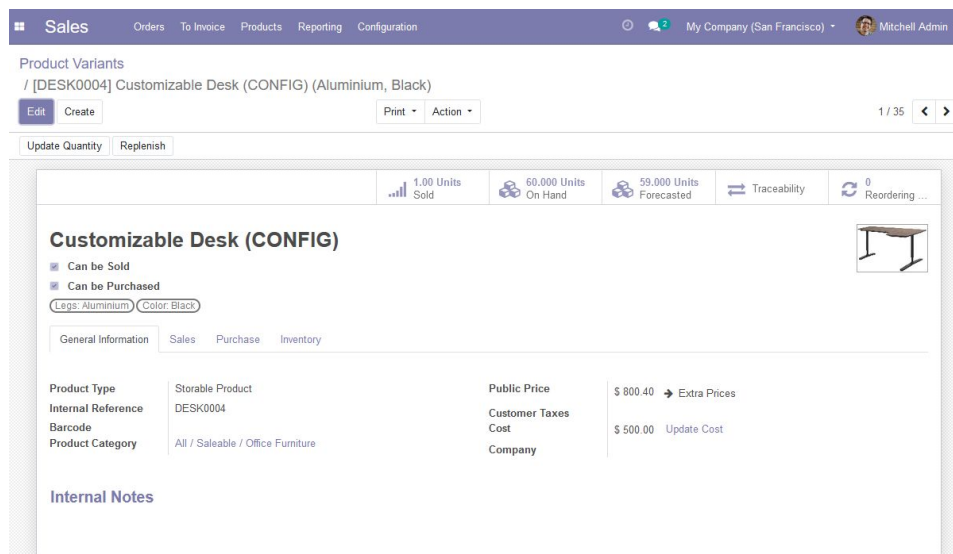
Captura 11

Caldrà afegir el següent:

- Una casella per indicar si el producte és de tipus **PG**.
- Un desplegable per indicar la tipologia del PG (**m3, m2, ml, pal**). Aquest desplegable estarà ocult si la casella de PG està desactivada.
- Una casella per indicar si el producte és **PVM**.
- Una casella per indicar si el producte és **PVC**.
- Un camp per indicar el **coeficient de conversió** entre l'UoM del producte i la fusta emprada. Aquest camp estarà ocult si la casella PVC està desactivada.
- Un camp per indicar quins **tractaments** accepta el producte.
- Una casella per indicar si el producte és un **tractament**.

Variant de producte

Es tracta d'una vista heretada que té l'aspecte de la Captura 12:



Captura 12

Caldrà afegir la informació de les plantilles de producte a més de la següent:

- Una casella per indicar si la variant té **dimensions per defecte**.
- Tres camps per indicar **cadascuna de les tres dimensions per defecte**. No seran visibles en cas que la casella que indica si té dimensions per defecte estiga desactivada.
- Una casella per indicar si les **dimensions per defecte estan bloquejades**. Aquesta està desactivada si la casella que indica si té dimensions per defecte ho està.

Matrius de preus de productes genèrics

Es tracta d'una vista nova que haurà de tindre els següents elements:

- **Productes** en què s'aplica la matriu.
- **Pricelists** en què s'aplica la matriu.
- **Unitat secundària** de la matriu
- Línies de matriu de preus de PG amb una columna per cadascuna d'aquestes dades: **llarg, gruix, ample, preu per unitat secundària i preu unitari de la peça**.

Introducció dels tractaments dels productes a les vendes

Es tracta d'una vista nova que haurà de tindre els següents elements:

- Un seleccionable dels **tractaments**.
- Un camp per introduir el **preu**.
- Un **seleccionable** per triar a quins productes de la venda se'ls vol aplicar els tractaments.

4.3.2. Característiques del sistema

En aquest apartat s'estableixen els requisits funcionals del sistema dividits per característiques, que es corresponen amb les funcionalitats establides a l'apartat 4.2.2. S'inclou també una estimació del cost de cada requisit utilitzant els valors 1, 2, 3, 5 i 8, proposats per Mike Cohn, i s'indica la prioritat amb les següents possibilitats⁵:

- **Must have.** El sistema ha de complir obligatòriament aquest requisit.
- **Should have.** Caldria que el sistema complira aquest requisit i s'ha de fer tot el possible per aconseguir-ho.
- **Could have.** Seria interessant que el sistema complira aquest requisit, però es podria prescindir d'ell.
- **Won't have.** El requisit no es complirà.

⁵ PRADEL MIQUEL, Jordi i RAYA MARTOS, Jose. *Gestió de requisits* [recurs d'aprenentatge]. Barcelona: UOC

4.3.2.1. Tipologia de productes

Objectiu de la característica

Els requisits sota aquesta característica tenen l'objectiu de permetre a l'usuari indicar que un producte concret és d'una tipologia especial de les presentades al llarg del treball així com especificar els seus trets definitoris. També donar les dimensions específiques als PG quan s'utilitzen a les vendes, albarans i factures.

Requisits funcionals

Id	Descripció	Cost	Prioritat
R-001	Definir productes PG de tipus m3	1	Must have
R-002	Definir productes PG de tipus m2	1	Must have
R-003	Definir productes PG de tipus ml	1	Must have
R-004	Definir productes PG de tipus pal	1	Must have
R-005	Indicar dimensions per defecte en PG	3	Could have
R-006	Poder bloquejar les dimensions per defecte en PG perquè no es puguin modificar en vendes, albarans i factures	3	Could have
R-007	Definir productes PVM	1	Must have
R-008	Definir productes PVC amb un coeficient	1	Should have ⁶
R-009	Indicar les dimensions dels PG en vendes, albarans i factures	1	Must have

⁶ S'ha indicat com a should have perquè sempre es podria introduir el volum de fusta d'aquest tipus de producte manualment definint-los com a PVM.

4.3.2.2. Càlcul de la fusta emprada i del volum, superfície o longitud dels PG

Objectiu de la característica

Els requisits recollits en aquesta subsecció tenen l'objectiu d'aconseguir calcular quanta fusta s'ha emprat als productes presents a les línies així com els m3, m2 o ml que representa cada element.

Requisits funcionals

Id	Descripció	Cost	Prioritat
R-010.1	Calcular els m3 en PG de tipus m3 a partir de les seues dimensions en vendes, factures i albarans	3	Must have
R-010.2	Calcular els m2 en PG de tipus m2 a partir de les seues dimensions en vendes, factures i albarans	3	Must have
R-010.3	Calcular els ml en PG de tipus ml a partir de les seues dimensions en vendes, factures i albarans	3	Must have
R-010.4	Calcular els ml en PG de tipus pal a partir de les seues dimensions en vendes, factures i albarans	3	Must have
R-011	Calcular el volum de fusta emprada dels productes genèrics en vendes, factures i albarans	3	Must have
R-012	Indicar en productes PVM el volum de fusta emprada manualment en vendes, factures i albarans	1	Must have
R-013	Calcular el volum de fusta emprada en productes PVC a partir de la introducció de la quantitat en vendes, factures i albarans	2	Should have
R-014	Calcular el volum de fusta total emprada en vendes, factures i albarans a través dels PG, PVM i PVC	1	Must have

4.3.2.3. Càlcul dels preus dels PG

Objectiu de la característica

Els requisits d'aquesta funcionalitat estan adreçats a l'establiment de preus dels PG. Com que aquests productes poden tindre diferents dimensions, el seu preu dependrà d'aquests valors. És a dir, un producte PG no tindrà un preu únic, sinó que aquest dependrà de les dimensions especificades a les vendes i factures. Dependran, així mateix, de la tarifa indicada.

Requisits funcionals

Id	Descripció	Cost	Prioritat
R-015	Calcular el preu unitari dels PG a partir del preu per unitat secundària (m3, m2 o ml) i les seues dimensions	5	Should have
R-016	Modificar manualment el preu de qualsevol producte en vendes i factures	1	Must have
R-017	Establir diferents preus per a tot tipus de productes segons el client a través de <i>pricelists</i>	3	Could have
R-018	Manipular el preu per unitat secundària dels PG per recalculer el seu preu unitari a les vendes i factures	3	Could have
R-019	Manipular el preu unitari dels PG per recalculer el seu preu per unitat secundària a les vendes i factures	3	Could have

4.3.2.4. Vendes, factures i albarans amb fusta PEFC

Objectiu de la característica

L'objectiu d'aquesta característica és indicar a les vendes, factures i albarans quan s'ha emprat fusta PEFC amb la finalitat de portar un registre de la fusta d'aquest tipus emprada així com saber amb exactitud quins són els productes demanats.

Requisits funcionals

Id	Descripció	Cost	Prioritat
R-020	Indicar en vendes, factures i albarans si la fusta emprada és PEFC	1	Should have
R-021	Filtrar vendes, factures i albarans segons si s'ha utilitzat o no fusta ecològica	1	Could have

4.3.2.5. Aplicació de tractaments als productes

Objectiu de la característica

Amb els requisits d'aquesta funcionalitat es persegueix poder indicar a les vendes si s'apliquen tractaments als productes seleccionats i quins tractaments s'apliquen. Els tractaments faran que el preu es veja afectat i la seua inclusió permetrà especificar quines característiques concretes han de tindre els productes que es lliuren al client.

Requisits funcionals

Id	Descripció	Cost	Prioritat
R-022	Indicar en els productes els tractaments que accepten o si no n'accepten cap	2	Could have
R-023	Aplicar tractaments als productes que n'accepten a les vendes	3	Must have
R-024	Indicar manualment el preu dels tractaments aplicats als productes a les vendes	1	Must have
R-025	Indicar als albarans quins tractaments s'han aplicat als productes a la venda des de la qual s'ha generat l'albarà	3	Should have
R-026	Indicar a les factures els tractaments que s'han aplicat als productes des de la venda de la qual procedeixen	3	Should have
R-027	Aplicar tractaments als productes que n'accepten a les factures	3	Won't have
R-028	Aplicar tractaments als productes que n'accepten als albarans	3	Won't have

4.3.2.6. Generació de documents amb nova informació

Objectiu de la característica

Odoo permet la generació de documents PDF amb la informació continguda a les instàncies d'alguns dels seus models, entre ells, les vendes, les factures i els albarans. L'objectiu d'aquesta funcionalitat és incloure la informació rellevant afegida a través dels mòduls desenvolupats.

Requisits funcionals

Id	Descripció	Cost	Prioritat
R-029	Mostrar als informes de vendes, factures i albarans les dimensions dels PG, els m3, m2 o ml (segons el tipus de producte)	8	Must have
R-030	Mostrar als informes de vendes, factures i albarans els tractaments aplicats als productes	5	Must have
R-031	Mostrar als informes de vendes, factures i albarans una imatge i un text al peu per indicar que la fusta emprada és de tipus PEFC	5	Must have

4.3.3. Requisits de rendiment

- El processament i el *feedback* del sistema han de ser immediats amb la realització dels següents càlculs:
 - Fusta emprada en una línia de venda, albarà o factura.
 - Fusta total emprada en una venda, albarà o factura.
 - El preu unitari d'un PG a la matriu de preus a partir de les seues dimensions i el preu per unitat secundària.
 - El preu unitari d'un PG en una línia de venda o factura quan es modifica el preu per unitat secundària.
 - El preu per unitat secundària d'un PG en una línia de venda o factura quan es modifica el preu unitari
- El sistema no podrà quedar bloquejat quan se seleccione un PG amb unes dimensions que tenen un preu assignat. És a dir, el preu haurà d'aparèixer a la línia immediatament.
- La generació dels informes en PDF amb sis productes amb la informació addicional d'aquests mòduls no podrà trigar més dos segons addicionals que la generació d'aquests informes sense que els mòduls estiguen instal·lats.

4.3.4. Restriccions de disseny

- Les línies de vendes i factures han d'aparèixer amb un color diferent quan continguem un PG amb unes dimensions especificades que no tenen un preu assignat.
- Els informes en PDF no hauran de contindre les columnes pròpies dels PG si no n'hi ha cap línia amb un PG.
- El nombre de xifres decimals que es mostren han de ser les següents:
 - 1 xifra decimal a l'ample i el gruix de les línies de venda, factura i albarà i dels seus respectius informes.
 - 2 xifres decimals al llarg de les línies de venda, factura i albarà i dels seus informes.
 - 3 xifres decimals als m3 dels PG de tipus m3 als informes de venda, albarà i factura.
 - 2 xifres decimals als m2 dels PG de tipus m2 i als ml dels PG de tipus ml i pal als informes de venda, albarà i factura.
 - 1 decimal a la columna de quantitat als informes de venda, albarà i factura.
- Els tractaments hauran de tindre una visualització diferent de les línies de productes normals a les vendes i factures.

4.3.5. Atributs del sistema de software

Les diferents característiques especificades s'han de desenvolupar en diferents mòduls amb l'objectiu que cadascun d'ells encapsule una sèrie de funcionalitats i que puguem funcionar de manera autònoma si és possible. És cert que el fet que es produïska un acoblament entre els mòduls és inevitable. Si un mòdul conté les funcionalitats relacionades amb la definició dels productes, el mòdul encarregat de calcular els preus a partir de les dimensions necessitarà al primer. El mòdul dels preus no podria funcionar de manera independent, però l'encarregat dels productes sí.

Així mateix, s'ha de garantir la futura extensió dels mòduls amb la incorporació de noves funcionalitats, ja estiguen contingudes dins dels mòduls desenvolupats en aquest projecte com en mòduls totalment nous.

De les qüestions de seguretat ja s'encarrega Odoo de base, però s'hauran de crear normes d'accés per als nous models, perquè només un usuari administrador en tinga accés.

4.3.6. Altres requisits no funcionals

Les noves dades introduïdes amb aquest desenvolupament s'hauran de traspasar de la venda cap a l'albarà i les factures.

4.4. Producte mínim lliurable

Conté els requisits amb una prioritat 'Must have' i 'Should have', que són els següents:

- Definir productes PG de tipus m3
- Definir productes PG de tipus m2
- Definir productes PG de tipus ml
- Definir productes PG de tipus pal
- Definir productes PVM
- Definir productes PVC amb un coeficient
- Indicar les dimensions dels PG en vendes, albarans i factures
- Calcular els m3 en PG de tipus m3 a partir de les seues dimensions en vendes, factures i albarans
- Calcular els m2 en PG de tipus m2 a partir de les seues dimensions en vendes, factures i albarans
- Calcular els ml en PG de tipus ml a partir de les seues dimensions en vendes, factures i albarans
- Calcular els ml en PG de tipus pal a partir de les seues dimensions en vendes, factures i albarans
- Calcular el volum de fusta emprada dels productes genèrics en vendes, factures i albarans
- Indicar en productes PVM el volum de fusta emprada en vendes, factures i albarans
- Calcular el volum de fusta emprada en productes PVC a partir de la introducció de la quantitat en vendes, factures i albarans
- Calcular el volum de fusta total emprada en vendes, factures i albarans a través dels PG, PVM i PVC
- Calcular el preu unitari dels PG a partir del preu per unitat secundària (m3, m2 o ml) i les seues dimensions
- Modificar manualment el preu de qualsevol producte en vendes i factures
- Aplicar tractaments des de les vendes als productes que n'accepten
- Indicar manualment el preu dels tractaments aplicats als productes
- Indicar als albarans quins tractaments s'han aplicat als productes a la venda des de la qual s'ha generat l'albarà
- Indicar a les factures els tractaments que s'han aplicat als productes des de la venda de la qual procedeixen
- Indicar en vendes, factures i albarans si la fusta emprada és PEFC
- Mostrar als informes de vendes, factures i albarans les dimensions dels PG i els m3, m2 o ml (segons el tipus de producte)
- Mostrar als informes de vendes, factures i albarans els tractaments aplicats als productes
- Mostrar als informes de vendes, factures i albarans una imatge i un text al peu per indicar que la fusta emprada és de tipus PEFC

Una vegada aconseguits tots els requisits inclosos al producte mínim lliurable, la selecció dels requisits posteriors es faria triant aquells que tenen un major nivell de prioritat. Entre aquests, es triarien aquells amb un cost inferior per anar avançant en l'assoliment de requisits.

4.5. Tests d'acceptació

Una vegada establits els requisits del sistema i definit el producte mínim lliurable, es presenta una sèrie de tests d'acceptació, seguint alguns dels principis del desenvolupament dirigit per tests (TDD). Aquest paradigma es pot definir com un procés de desenvolupament de programari basat en la conversió dels requisits del *software* en casos de prova abans que el programari siga completament desenvolupat i el monitoratge de tot el desenvolupament a través del testatge continuat amb aquests casos de prova⁷.

Carlo Blé, al seu llibre *Disseny àgil amb TDD*, esmena, entre els pilars fonamentals del TDD, dos que s'han considerat molt interessants per assumir la base d'aquesta metodologia: la implementació de les funcions justes que el client necessita i res més i la minimització del nombre de defectes que arriben al *software* en fase de producció⁸. L'autor proposa que cada història d'usuari tinga associats una sèrie de tests que permeten comprovar que el *software* compleix amb els requisits recollits.

Els tests es documenten a través de casos de prova, definits a la norma IEEE/ISO/IEC 29119 com un conjunt de precondicions, entrades i resultats esperats dissenyats per a conduir l'execució dels tests amb l'objectiu d'assolir els objectius marcats⁹. Seguint aquest mateix estàndard, la documentació de cadascun dels casos de prova estarà composta d'un identificador únic, l'objectiu, unes precondicions (en forma d'altres proves que s'hauran d'haver realitzat prèviament o d'accions necessàries), uns *inputs* i els resultats esperats. En aquesta memòria apareixen organitzats segons històries d'usuari, seguint l'exposat per Carlo Blé.

En haver adoptat un paradigma de desenvolupament de projecte basat en el disseny de proves en una fase primerenca, s'han hagut d'introduir conceptes relacionats amb la gestió de qualitat, que es reprendran en l'últim apartat de la memòria.

⁷ *Testdriven development* [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: https://en.wikipedia.org/wiki/Test-driven_development

⁸ BLÉ JURADO, Carlos. *Diseño Ágil con TDD*. iExpertos, 2010. ISBN: 978-1445264714

⁹ INTERNATIONAL ORGANIZATION FOR STANDARIZATION. *Software and systems engineering*. IEEE/ISO/IEC 29119. 2009

Com a gestor vull definir el tipus de producte

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR001	Definir PG tipus m3.		Crear un producte 'Biga' amb l'opció PG marcada i el subtipus m3 seleccionat. Triar 'Unitats' com a UoM.	S'introdueix una nova plantilla de producte PG en m3 a la base de dades amb nom 'Biga'. Es crea una variant amb nom 'Biga' amb les dades sincronitzades amb la <i>template</i> .
PR002	Definir PG tipus m2.		Crear un producte 'Tauler' amb l'opció PG marcada i el subtipus m2 seleccionat. Triar 'Unitats' com a UoM.	S'introdueix una nova plantilla de producte PG en m2 a la base de dades amb nom 'Tauler'. Es crea una variant amb nom 'Tauler' amb les dades sincronitzades amb la <i>template</i> .
PR003	Definir PG tipus ml.		Crear un producte 'Llistó' amb l'opció PG marcada i el subtipus ml seleccionat. Triar 'Unitats' com a UoM.	S'introdueix una nova plantilla de producte PG en ml a la base de dades amb nom 'Llistó'. Es crea una variant amb nom 'Llistó' amb les dades sincronitzades amb la <i>template</i> .
PR004	Definir PG tipus pal.		Crear un producte 'Barilla' amb l'opció PG marcada i el subtipus pal seleccionat. Triar 'Unitats' com a UoM.	S'introdueix una nova plantilla de producte PG de tipus pal a la base de dades amb nom 'Barilla'. Es crea una variant amb nom 'Barilla' amb les dades sincronitzades amb la <i>template</i> .
PR005	Definir un PVC.		Crear un producte 'Serradures' amb l'opció PVC marcada, amb la UoM 'KG' i coeficient 1,25.	S'introdueix una nova plantilla de producte PVC a la base de dades amb nom 'Serradures' amb coeficient de conversió 1,25. Es crea una variant amb nom 'Serradures' amb les dades sincronitzades amb la <i>template</i> .
PR006	Definir un PVM.		Crear el producte 'Troncs' amb l'opció PVM seleccionada.	S'introdueix una nova plantilla de producte PVM a la base de dades amb nom 'Troncs'. Es crea una variant amb nom 'Troncs' amb les dades sincronitzades amb la <i>template</i> .
PR007	Seleccionar un sol tipus de producte.		Al formulari d'un producte, intentar seleccionar dos tipus de producte (d'entre PG, PVC i PVM).	No és possible seleccionar dos tipus de producte.

PR008	Seleccionar un sol subtipus de PG.		Al formulari d'un producte, amb l'opció PG seleccionada, intentar seleccionar dos subtipus de PG.	No és possible seleccionar dos subtipus de PG.
PR009	Seleccionar un coeficient de conversió només si el producte és PVC.		Intentar introduir un coeficient de conversió en un producte amb l'opció PVC desmarcada.	No és possible introduir un coeficient en un producte amb l'opció PVC desmarcada.
PR010 ¹⁰	Definir coeficients només amb valors positius.		Introduir el coeficient -1 a un producte PVC.	Apareix un avís que indica que no es pot establir un valor negatiu en el camp de coeficient.

Com a gestor vull un càlcul de la fusta emprada, els m3, m2 i ml dels PG a les vendes, factures i albarans¹¹.

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR011	Calcular els m3 i la fusta emprada dels PG tipus m3 a vendes, albarans i factures amb quantitat 1.	PR001	Seleccionar el producte 'Biga' a una venda/albarà/factura. Establir les dimensions llarg=1, ample=30 i alt=30. Establir 1 unitat.	Columna m3: 0,09 (només lectura) Columna volum fusta: 0,09 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 0,09 (només lectura)
PR012	Calcular els m3 i la fusta emprada dels PG tipus m3 a vendes, albarans i factures amb quantitat > 1.	PR001	Seleccionar el producte 'Biga' a una venda/albarà/factura. Establir les dimensions llarg=1, ample=30 i alt=30. Establir 2 unitats.	Columna m3: 0,18 (només lectura) Columna volum fusta: 0,18 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 0,18 (només lectura)
PR013	Calcular els m3 i la fusta emprada dels PG tipus m3 a vendes, albarans i factures amb UoM diferent de la triada per defecte i quantitat > 1.	PR001	Seleccionar el producte 'Biga' a una venda/albarà/factura. Establir les dimensions llarg=1, ample=30 i alt=30. Establir 2 dotzenes.	Columna m3: 2,16 (només lectura) Columna volum fusta: 2,16 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 2,16 (només lectura)

¹⁰ Es podria afegir un cas frontera amb el valor de coeficient 0, que sí hauria de ser vàlid.

¹¹ A tot el sistema, llarg es defineix en metres i ample i alt en centímetres.

PR014	Calcular els m2 i la fusta emprada dels PG tipus m2 a vendes, albarans i factures amb quantitat 1.	PR002	Seleccionar el producte 'Tauler' a una venda, albarà o factura. Establir les dimensions llarg=1, ample=20 i alt=1. Establir 1 unitat.	Columna m2: 0,2 (només lectura) Columna volum fusta: 0,002 (només lectura) Columnes m3 i ml desactivades m3 totals venda/albarà/factura: 0,002 (només lectura)
PR015	Calcular els m2 i la fusta emprada dels PG tipus m2 a vendes, albarans i factures amb quantitat > 1.	PR002	Seleccionar el producte 'Tauler' a una venda, albarà o factura. Establir les dimensions llarg=1, ample=20 i alt=1. Establir 2 unitats.	Columna m2: 0,4 (només lectura) Columna volum fusta: 0,004 (només lectura) Columnes m3 i ml desactivades m3 totals venda/albarà/factura: 0,004 (només lectura)
PR016	Calcular els m2 i la fusta emprada dels PG tipus m2 a vendes, albarans i factures amb UoM diferent de la triada per defecte i quantitat > 2.	PR002	Seleccionar el producte 'Tauler' a una venda, albarà, o factura. Establir les dimensions llarg=1, ample=20 i alt=1. Establir 2 dotzenes.	Columna m2: 4,8 (només lectura) Columna volum fusta: 0,048 (només lectura) Columnes m3 i ml desactivades m3 totals venda/albarà/factura: 0,048 (només lectura)
PR017	Calcular els ml i la fusta emprada dels PG tipus ml a vendes, albarans i factures amb quantitat 1.	PR003	Seleccionar el producte 'Llistó' a una venda, albarà o factura. Establir les dimensions llarg=2, ample=5 i alt=1. Establir 1 unitat.	Columna ml: 2. Columna volum fusta: 0,001. Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,001
PR018	Calcular els ml i la fusta emprada dels PG tipus ml a vendes, albarans i factures amb quantitat > 1.	PR003	Seleccionar el producte 'Llistó' a una venda, albarà o factura. Establir les dimensions llarg=2, ample=5 i alt=1. Establir 2 unitats.	Columna ml: 4 (només lectura) Columna volum fusta: 0,002 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,002 (només lectura)
PR019	Calcular els ml i la fusta emprada dels PG tipus ml a vendes, albarans i factures amb UoM diferent de la triada per defecte i quantitat > 2.	PR003	Seleccionar el producte Llistó a una venda/albarà/factura. Establir les dimensions llarg=2, ample=5 i alt=1. Establir 2 dotzenes.	Columna ml: 48 (només lectura) Columna volum fusta: 0,024 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,024 (només lectura)

PR020	Calcular els ml i la fusta emprada dels PG tipus pal a vendes, albarans i factures amb quantitat 1.	PR004	Seleccionar el producte 'Barilla' a una venda, albarà o factura. Establir les dimensions llarg=2, ample=5 i alt=4. Establir 1 unitat.	Columna ml: 2 (només lectura) Columna volum fusta: 0,003 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,003 (només lectura)
PR021	Calcular els ml i la fusta emprada dels PG tipus pal a vendes, albarans i factures amb quantitat > 1.	PR004	Seleccionar el producte 'Barilla' a una venda, albarà o factura. Establir les dimensions llarg=2, ample=5 i alt=4 Establir 2 unitats	Columna ml: 2 (només lectura) Columna volum fusta: 0,006 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,006 (només lectura)
PR022	Calcular els ml i la fusta emprada dels PG tipus pal a vendes, albarans i factures amb UoM diferent de la triada per defecte i quantitat > 1.	PR004	Seleccionar el producte 'Barilla' a una venda, albarà o factura. Establir les dimensions llarg=2, ample=5 i alt=4. Establir 2 dotzenes.	Columna ml: 48 (només lectura) Columna volum fusta: 0,076 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,076 (només lectura)
PR023	L'ample dels PG pal ha de ser major que l'alt a les vendes albarans i factures	PR004	Seleccionar el producte 'Barilla' a una venda, albarà, o factura. Establir les dimensions ample=1 alt=2.	Es mostra un error que indica que l'ample ha de ser més gran que l'alt amb els PG de tipus pal.
PR024	Calcular els m3 dels productes PVC a una venda, albarà o factura amb la unitat de mesura per defecte del producte.	PR005	Seleccionar el producte 'Serradures' a una venda, albarà o factura. Establir la quantitat a 25 kg.	Columna m3: 31,25 (només lectura) Columna volum fusta: 31,25 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 31,25 (només lectura)
PR025	Calcular els m3 dels productes PVC a una venda, albarà o factura amb la UoM diferent de l'establida per defecte al producte.	PR005	Seleccionar el producte 'Serradures' a una venda, albarà o factura. Establir la quantitat a 1 tona.	Columna m3: 1250 (només lectura) Columna volum fusta: 1250 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 1250 (només lectura)
PR026	Calcular la fusta emprada per productes PVM.	PR006	Seleccionar el producte 'Troncs' a una venda, albarà o factura. Establir el valor 2 a la columna m3.	Columna m3: 2 Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 2 (només lectura)

Com a gestor vull poder definir les dimensions per defecte dels PG

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR027	Definir dimensions per defecte dels PG de tipus m3.		Crear el producte 'Biga 2-30x30' com a PG de tipus m3. Assignar dimensions per defecte llarg=2, ample=30 i alt=30.	S'introdueix una nova plantilla de producte PG en m3 a la base de dades amb nom 'Biga 2-30x30'. Es crea una variant amb nom 'Biga 2-30x30' amb les dades sincronitzades amb la <i>template</i> .
PR028	Definir dimensions per defecte dels PG de tipus m2.		Crear el producte 'Tauler 1-20' com a PG de tipus m2. Assignar dimensions per defecte llarg=1, ample=20 i alt=1.	S'introdueix una nova plantilla de producte PG en m2 a la base de dades amb nom 'Tauler 1-20'. Es crea una variant amb nom 'Tauler 1-20' amb les dades sincronitzades amb la <i>template</i> .
PR029	Definir dimensions per defecte dels PG de tipus ml.		Crear el producte 'Llistó 2' com a PG de tipus ml. Assignar dimensions per defecte llarg=2, ample=5 i alt=1.	S'introdueix una nova plantilla de producte PG en ml a la base de dades amb nom 'Llistó 2'. Es crea una variant amb nom 'Llistó 2' amb les dades sincronitzades amb la <i>template</i> .
PR030	Definir dimensions per defecte de PG de tipus pal.		Crear el producte 'Barilla 2' com a PG de tipus pal. Assignar dimensions per defecte llarg=2, ample=5 i alt=4.	S'introdueix una nova plantilla de producte PG de tipus pal a la base de dades amb nom 'Barilla 2'. Es crea una variant amb nom 'Barilla 2' amb les dades sincronitzades amb la <i>template</i> .
PR031	Introduir un PG en m3 amb dimensions per defecte a una venda, albarà o factura amb quantitat 1.	PR027	Seleccionar el producte 'Biga 2-30x30' a una venda/albarà/factura. Establir la quantitat a 1 unitat.	Columna llarg: 2 Columna ample: 30 Columna alt: 30 Columna m3: 0,18 (només lectura) Columna volum fusta: 0,18 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 0,18 (només lectura)
PR032	Introduir un PG en m3 amb dimensions per defecte a una venda, albarà o factura amb quantitat > 1	PR027	Seleccionar el producte 'Biga 2-30x30' a una venda/albarà/factura. Establir la quantitat a 2 unitats.	Columna llarg: 2 Columna ample: 30 Columna alt: 30 Columna m3: 0,36 (només lectura) Columna volum fusta: 0,36 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 0,36 (només lectura)

PR033	Introduir un PG en m3 amb dimensions per defecte a una venda, albarà o factura amb UoM diferent de la triada per defecte i quantitat > 1.	PR027	Seleccionar el producte 'Biga 2-30x30' a una venda/albarà/factura. Establir la quantitat a 2 dotzenes.	Columna llarg: 2 Columna ample: 30 Columna alt: 30 Columna m3: 4,32 (només lectura) Columna volum fusta: 4,32 (només lectura) Columnes m2 i ml desactivades m3 totals venda/albarà/factura: 4,32 (només lectura)
PR034	Introduir un PG en m2 amb dimensions per defecte a una venda, albarà o factura amb quantitat 1.	PR028	Seleccionar el producte 'Tauler 1-20' a una venda, albarà o factura. Establir la quantitat a 1 unitat.	Columna llarg: 1 Columna ample: 20 Columna alt: 1 Columna m2: 0,2 (només lectura) Columna volum fusta: 0,002 (només lectura) Columnes m3 i ml desactivades m3 totals venda/albarà/factura: 0,002 (només lectura)
PR035	Introduir un PG en m2 amb dimensions per defecte a una venda, albarà o factura amb quantitat > 1.	PR028	Seleccionar el producte 'Tauler 1-20' a una venda, albarà o factura. Establir la quantitat a 2 unitats.	Columna llarg: 1 Columna ample: 20 Columna alt: 1 Columna m2: 0,4 (només lectura) Columna volum fusta: 0,004 (només lectura) Columnes m3 i ml desactivades m3 totals venda/albarà/factura: 0,004 (només lectura)
PR036	Introduir un PG en m2 amb dimensions per defecte a una venda, albarà o factura amb UoM diferent de la triada per defecte i quantitat > 1.	PR028	Seleccionar el producte 'Tauler 1-20' a una venda, albarà o factura. Establir la quantitat a 2 dotzenes.	Columna llarg: 1 Columna ample: 20 Columna alt: 1 Columna m2: 4,8 (només lectura) Columna volum fusta: 0,048 (només lectura) Columnes m3 i ml desactivades m3 totals venda/albarà/factura: 0,048 (només lectura)
PR037	Introduir un PG en ml amb dimensions per defecte a una venda, albarà o factura amb quantitat 1.	PR029	Seleccionar el producte 'Llistó 2' a una venda, albarà o factura. Establir la quantitat a 1 unitat.	Columna llarg: 2 Columna ample: 5 Columna alt: 1 Columna ml: 2 (només lectura) Columna volum fusta: 0,001 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,001 (només lectura)

PR038	Introduir un PG en ml amb dimensions per defecte a una venda, albarà o factura amb quantitat > 1.	PR029	Seleccionar el producte 'Llistó 2' a venda, albarà o factura. Establir la quantitat a 2 unitats.	Columna llarg: 2 Columna ample: 5 Columna alt: 1 Columna ml: 4 (només lectura) Columna volum fusta: 0,002 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,002 (només lectura)
PR039	Introduir un PG en ml amb dimensions per defecte a una venda, albarà o factura amb UoM diferent de la triada per defecte i quantitat > 1.	PR029	Seleccionar el producte 'Llistó 2' a venda, albarà o factura. Establir la quantitat a 2 dotzenes.	Columna llarg: 2 Columna ample: 5 Columna alt: 1 Columna ml: 48 (només lectura) Columna volum fusta: 0,024 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,024 (només lectura)
PR040	Introduir un PG pal amb dimensions per defecte a una venda, albarà o factura amb quantitat 1.	PR030	Seleccionar el producte 'Barilla 2' a una venda, albarà o factura. Establir la quantitat a 1 unitat.	Columna llarg: 2 Columna ample: 5 Columna alt: 4 Columna ml: 2 (només lectura) Columna volum fusta: 0,003 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,003 (només lectura)
PR041	Introduir un PG pal amb dimensions per defecte a una venda, albarà o factura amb quantitat > 1.	PR030	Seleccionar el producte 'Barilla 2' a venda, albarà o factura. Establir la quantitat a 2 unitats.	Columna llarg: 2 Columna ample: 5 Columna alt: 4 Columna ml: 4 (només lectura) Columna volum fusta: 0,006 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,006 (només lectura)
PR042	Introduir un PG pal amb dimensions per defecte a una venda, albarà o factura amb UoM diferent de la triada per defecte i quantitat > 1.	PR030	Seleccionar el producte 'Barilla 2' a venda, albarà o factura. Establir la quantitat a 2 dotzenes.	Columna llarg: 2 Columna ample: 5 Columna alt: 4 Columna ml: 48 (només lectura) Columna volum fusta: 0,076 (només lectura) Columnes m3 i m2 desactivades m3 totals venda/albarà/factura: 0,076 (només lectura)
PR043	Introduir valors per defecte als PG de tipus pal amb ample major que alt.		Crear un PG de tipus pal amb els valors per defecte llarg > 0, alt=2 i ample=1.	Es mostra un avís que indica que a les dimensions per defecte dels PG de tipus pal ample ha de ser major que alt.

PR044	El llarg per defecte dels PG ha de ser un valor positiu.		Crear un PG de qualsevol tipus amb llarg per defecte establert a -1.	Es mostra un avís que indica que el llarg per defecte dels PG ha de ser un valor positiu.
PR045	L'ample per defecte dels PG ha de ser un valor positiu.		Crear un PG de qualsevol tipus amb l'ample per defecte establert a -1.	Es mostra un avís que indica que l'ample per defecte dels PG ha de ser un valor positiu.
PR046	L'alt per defecte dels PG ha de ser un valor positiu.		Crear un PG de qualsevol tipus amb l'alt per defecte establert a -1.	Es mostra un avís que indica que l'alt per defecte dels PG ha de ser un valor positiu.

Com a gestor vull poder bloquejar les dimensions per defecte dels PG

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR047	Bloquejar les dimensions per defecte d'un PG en m3.	PR027	Seleccionar l'opció de bloquejar les dimensions per defecte del producte 'Biga 2-30x30'. Triar el producte 'Biga 2-30x30' a una venda, albarà o factura.	Columnes llarg, ample i alt: només lectura
PR048	Bloquejar les dimensions per defecte d'un PG en m2.	PR028	Seleccionar l'opció de bloquejar les dimensions per defecte del producte 'Tauler 1-20'. Triar el producte 'Tauler 1-20' a una venda, albarà o factura.	Columnes llarg, ample i alt: només lectura
PR049	Bloquejar les dimensions per defecte d'un PG en ml.	PR029	Seleccionar l'opció de bloquejar les dimensions per defecte del producte 'Llistó 2'. Triar el producte 'Llistó 2' a una venda, albarà o factura.	Columnes llarg, ample i alt: només lectura

PR050	Bloquejar les dimensions per defecte d'un PG de tipus pal.	PR030	Seleccionar l'opció de bloquejar les dimensions per defecte del producte 'Barilla 2'. Triar el producte 'Barilla 2' a una venda, albarà o factura.	Columnes llarg, ample i alt: només lectura
PR051	Triar l'opció de bloquejar dimensions per defecte només en productes PG.		Des de la pantalla de creació/edició de productes, seleccionar que el producte no siga PG.	No s'ha de mostrar l'opció de bloquejar les dimensions per defecte.

Com a gestor vull modificar les característiques configurables dels PG, PVM i PVC

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR052	Modificar el tipus de producte d'una plantilla amb només la variant creada per defecte.	PR001 PR002 PR003 PR004 PR005 PR006	Editar el tipus d'una <i>template</i> PG, PVM o PVC amb variant única i seleccionar un altre tipus.	El tipus de producte s'ha modificat a la <i>template</i> . El tipus de producte s'ha canviat a la variant i coincideix amb el de la <i>template</i> .
PR053	Modificar el tipus de producte d'una variant única.	PR001 PR002 PR003 PR004 PR005 PR006	Editar el tipus d'una variant única PG, PVM o PVC i seleccionar un altre tipus.	El tipus de producte s'ha modificat a la variant. El tipus de producte s'ha canviat a la <i>template</i> i coincideix amb el de la variant.
PR054	Modificar el subtipus d'un PG d'una plantilla amb només la variant creada per defecte.	PR001 PR002 PR003 PR004	Editar el subtipus d'un PG amb variant única i seleccionar un altre subtipus.	El subtipus de producte s'ha modificat a la <i>template</i> . El subtipus de producte s'ha canviat a la variant i coincideix amb el de la <i>template</i> .
PR055	Modificar el subtipus de PG d'una variant única.	PR001 PR002 PR003 PR004	Editar el subtipus d'una variant única PG i seleccionar un altre subtipus.	El subtipus de producte s'ha modificat a la variant. El subtipus de producte s'ha canviat a la <i>template</i> i coincideix amb el de la variant.
PR056	Modificar el coeficient d'un PVC d'una plantilla amb només la variant creada per defecte.	PR005	Editar el coeficient des de la <i>template</i> d'un PVC amb variant única.	El coeficient s'ha modificat a la <i>template</i> . El coeficient s'ha canviat a la variant i coincideix amb el de la <i>template</i> .
PR057	Modificar el coeficient d'un PVC d'una variant única.	PR005	Editar el coeficient d'un PVC des d'una variant única.	El coeficient s'ha modificat a la variant. El coeficient s'ha canviat a la <i>template</i> i coincideix amb el de la variant.

PR058	Modificar el tipus de producte d'una plantilla amb diverses variants.	Crear un producte PG, PVC o PVM i assignar-li diverses variants.	Canviar el tipus de producte des de la plantilla.	El tipus s'ha modificat a la plantilla. El tipus s'ha modificat a cadascuna de les variants i coincideix amb el de la <i>template</i> .
PR059	Modificar el subtipus d'un PG d'una plantilla amb diverses variants.	Crear un producte PG de qualsevol subtipus i assignar-li diverses variants.	Canviar el subtipus des de la plantilla.	El subtipus s'ha modificat a la plantilla. El subtipus s'ha modificat a cadascuna de les variants i coincideix amb el de la <i>template</i> .
PR060	Seleccionar dimensions per defecte dels PG només des de les variants quan la <i>template</i> té diverses variants.	PR059	Editar la plantilla d'un PG amb diverses variants.	No han d'aparèixer les dimensions per defecte a la plantilla. S'ha de mostrar un missatge que indique que les dimensions per defecte es gestionen des de les variants.
PR061	Seleccionar l'opció de bloquejar les dimensions per defecte dels PG només des de les variants quan la <i>template</i> té diverses variants.	PR059	Editar la plantilla d'un PG amb diverses variants.	No ha d'aparèixer l'opció de bloquejar les dimensions per defecte a la plantilla. S'ha de mostrar un missatge que indique que l'opció de bloqueig de les dimensions per defecte es gestiona des de cada variant.
PR062	Seleccionar el coeficient de conversió dels PVC només des de les variants quan la <i>template</i> té diverses variants.	Crear un PVC amb diverses variants.	Editar la plantilla del PVC creat.	No ha d'aparèixer el coeficient a la plantilla. S'ha de mostrar un missatge que indique que el coeficient es gestiona des de cada variant.
PR063	Canviar les dimensions per defecte des d'una de les múltiples variants d'un PG des de cadascuna de les variants.	PR059	Canviar les dimensions per defecte d'un PG amb diverses variants des de la variant.	Les dimensions per defecte només s'han modificat a la variant concreta.
PR064	Modificar l'opció de bloquejar les dimensions per defecte des d'una de les múltiples variants d'un PG des de cadascuna de les variants.	PR059	Seleccionar / Deseleccionar l'opció de bloquejar les dimensions per defecte d'un PG amb diverses variants des de la variant.	L'opció de bloquejar dimensions per defecte només s'ha modificat a la variant concreta.

Com a gestor vull definir els preus per unitat secundària dels PG segons les seues dimensions

NOTA: Cal tindre definida la pricelist 'Comandes de baix valor'.

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR065	Seleccionar només PG a les matrius de preus.	Tindre PG a la base de dades.	Crear una nova matriu de preus. Comprovar els productes seleccionables.	Només es poden seleccionar PG.
PR066	Seleccionar tots els productes d'una matriu de preus del mateix subtipus (m3).	PR001, PR002, PR003, PR004 Crear un segon PG en m3	Seleccionar el producte 'Biga' a una matriu de preus.	La matriu de preus mostra que la unitat secundària emprada són els m3. Només són seleccionables productes PG en m3.
PR067	Seleccionar tots els productes d'una matriu de preus del mateix subtipus (m2).	PR001, PR002, PR003, PR004 Crear un segon PG en m2.	Seleccionar el producte 'Tauler' a una matriu de preus.	La matriu de preus mostra que la unitat secundària emprada són els m2. Només són seleccionables productes PG en m2.
PR068	Seleccionar tots els productes d'una matriu de preus del mateix subtipus (ml).	PR001, PR002, PR003, PR004	Seleccionar el producte 'Llistó' a una matriu de preus.	La matriu de preus mostra que la unitat secundària emprada són els ml. Només són seleccionables productes PG en ml i de tipus pal.
PR069	Seleccionar tots els productes d'una matriu de preus del mateix subtipus (ml, amb tipus pal).	PR001, PR002, PR003, PR004	Seleccionar el producte 'Barilla' a una matriu de preus.	La matriu de preus mostra que la unitat secundària emprada són els ml. Només són seleccionables productes PG en ml i de tipus pal.
PR070	Seleccionar qualsevol producte PG quan s'han eliminat tots els productes a la matriu de preus.		Eliminar tots els productes d'una matriu de preus qualsevol.	La matriu de preus no mostra cap unitat secundària emprada. Són seleccionables tots els PG.
PR071	Calcular el preu unitari dels PG de tipus m3	PR001	Seleccionar el producte 'Biga' a una matriu de preus. Seleccionar la tarifa 'Public Pricelist'. Introduir una línia amb les dimensions llarg=2, ample=20 i alt=20 i 60 com a preu per unitat secundària.	Columna preu unitari: 4,8

PR072	Calcular el preu unitari dels PG de tipus m2	PR002	Seleccionar el producte 'Tauler' a una matriu de preus. Seleccionar la tarifa 'Public Pricelist'. Introduir una línia amb les dimensions llarg=1, ample=50 i alt=2 i 10 com a preu per unitat secundària.	Columna preu unitari: 5
PR073	Calcular el preu unitari dels PG de tipus ml	PR003	Seleccionar el producte 'Llistó' a una matriu de preus. Seleccionar la tarifa 'Public Pricelist'. Introduir una línia amb les dimensions llarg=1, ample=2 i alt=1 i 3 com a preu per unitat secundària.	Columna preu unitari: 3
PR074	Calcular el preu unitari dels PG de tipus pal	PR004	Seleccionar el producte 'Barilla' a una matriu de preus. Seleccionar la tarifa 'Public Pricelist'. Introduir una línia amb les dimensions llarg=1, ample=2 i alt=1 i 3 com a preu per unitat secundària.	Columna preu unitari: 3
PR075	Introduir línies a la matriu de preus amb combinació única de dimensions.		Crear dues línies a la matriu de preus amb combinació de dimensions idèntiques.	Es mostra un avís que indica que la combinació de dimensions a una matriu ha de ser única. No es guarda la matriu.
PR076	Introduir línies amb el gruix major que l'ample a les matrius que continguin PG de tipus pal.	PR003, PR004	Introduir els productes 'Llistó' i 'Barilla' a una matriu de preus. Crear una línia a la matriu amb les dimensions llarg=1, ample=1 i alt=2.	Es mostra un avís que indica que a les matrius amb PG de tipus pal totes les línies han de complir que ample > gruix. No es guarda la matriu de preus.

PR077	Mantindre cada producte en només una matriu de preus vinculada a una pricelist.	PR001	<p>Crear una matriu de preus amb el producte 'Biga' vinculada a les pricelists 'Public Pricelist' (per defecte a Odoo) i Comandes de baixa valor'.</p> <p>Crear una segona matriu de preus amb el producte 'Biga' vinculada a la tarifa 'Public Pricelist'.</p>	<p>Apareix un avís que indica que el producte 'Biga' ja està present a una altra matriu de preus vinculada a la tarifa 'Public Pricelist'.</p> <p>No es guarda la segona matriu de preus.</p>
-------	---	-------	---	---

Com a gestor vull que els preus segons les dimensions dels PG apareguen a vendes i factures quan se seleccionen els productes amb les dimensions contemplades en matrius de preus

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR078	Mostrar preus dels PG en m3 configurats a les matrius a vendes i factures amb 1 establert com a quantitat.	PR071	<p>Introduir el producte 'Biga' a una venda o factura.</p> <p>Seleccionar la tarifa 'Public Pricelist'.</p> <p>Seleccionar 1 unitat.</p> <p>Introduir les dimensions llarg=2, ample=20 i alt=20</p>	<p>Columna preu per unitat secundària: 60</p> <p>Columna preu unitari: 4,8</p> <p>Columna subtotal: 4,8</p>
PR079	Mostrar preus dels PG en m3 configurats a les matrius a vendes i factures amb quantitat > 1.	PR071	<p>Introduir el producte 'Biga' a una venda o factura.</p> <p>Seleccionar la tarifa 'Public Pricelist'.</p> <p>Seleccionar 2 unitats.</p> <p>Introduir les dimensions llarg=2, ample=20 i alt=20</p>	<p>Columna preu per unitat secundària: 60</p> <p>Columna preu unitari: 4,8</p> <p>Columna subtotal: 9,6</p>
PR080	Mostrar preus dels PG en m3 configurats a les matrius a vendes i factures amb UoM diferent de l'establida al producte i quantitat > 1.	PR071	<p>Introduir el producte 'Biga' a una venda o factura.</p> <p>Seleccionar la tarifa 'Public Pricelist'.</p> <p>Seleccionar 2 dotzenes.</p> <p>Introduir les dimensions llarg=2, ample=20 i alt=20</p>	<p>Columna preu per unitat secundària: 60</p> <p>Columna preu unitari: 4,8</p> <p>Columna subtotal: 115,2</p>

PR081	Mostrar preus dels PG en m2 configurats a les matrius a vendes i factures amb 1 establert com a quantitat.	PR072	Introduir el producte 'Tauler' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 1 unitat. Introduir les dimensions llarg=1, ample=50 i alt=2	Columna preu per unitat secundària: 10 Columna preu unitari: 5 Columna subtotal: 5
PR082	Mostrar preus dels PG en m2 configurats a les matrius a vendes i factures amb quantitat > 1.	PR072	Introduir el producte 'Tauler' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 2 unitats. Introduir les dimensions llarg=1, ample=50 i alt=2	Columna preu per unitat secundària: 10 Columna preu unitari: 5 Columna subtotal: 10
PR083	Mostrar preus dels PG en m2 configurats a les matrius a vendes i factures amb UoM diferent de l'establida al producte i quantitat > 1.	PR072	Introduir el producte 'Tauler' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 2 dotzenes. Introduir les dimensions llarg=1, ample=50 i alt=2	Columna preu per unitat secundària: 10 Columna preu unitari: 5 Columna subtotal: 120
PR084	Mostrar preus dels PG en ml configurats a les matrius a vendes i factures amb 1 establert com a quantitat.	PR073	Introduir el producte 'Llistó' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 1 unitat. Introduir les dimensions llarg=1, ample=2 i alt=1	Columna preu per unitat secundària: 3 Columna preu unitari: 3 Columna subtotal: 3
PR085	Mostrar preus dels PG en ml configurats a les matrius a vendes i factures amb quantitat > 1.	PR073	Introduir el producte 'Llistó' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 2 unitats. Introduir les dimensions llarg=1, ample=2 i alt=1	Columna preu per unitat secundària: 3 Columna preu unitari: 3 Columna subtotal: 6

PR086	Mostrar preus dels PG en ml configurats a les matrius a vendes i factures amb UoM diferent de l'establida al producte i quantitat > 1.	PR073	Introduir el producte 'Llistó' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 2 dotzenes. Introduir les dimensions llarg=1, ample=2 i alt=1.	Columna preu per unitat secundària: 3 Columna preu unitari: 3 Columna subtotal: 72
PR087	Mostrar preus dels PG de tipus pal configurats a les matrius a vendes i factures amb 1 establert com a quantitat.	PR074	Introduir el producte 'Barilla' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 1 unitat. Introduir les dimensions llarg=1, ample=2 i alt=1	Columna preu per unitat secundària: 3 Columna preu unitari: 3 Columna subtotal: 3
PR088	Mostrar preus dels PG de tipus pal configurats a les matrius a vendes i factures amb quantitat > 1.	PR074	Introduir el producte 'Barilla' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 2 unitats. Introduir les dimensions llarg=1, ample=2 i alt=1	Columna preu per unitat secundària: 3 Columna preu unitari: 3 Columna subtotal: 6
PR089	Mostrar preus dels PG de tipus pal configurats a les matrius a vendes i factures amb UoM diferent de l'establida al producte i quantitat > 1.	PR074	Introduir el producte 'Barilla' a una venda o factura. Seleccionar la tarifa 'Public Pricelist'. Seleccionar 2 dotzenes. Introduir les dimensions llarg=1, ample=2 i alt=1	Columna preu per unitat secundària: 3 Columna preu unitari: 3 Columna subtotal: 72
PR90	Posar a 0 el preu unitari i per unitat secundària de les línies amb PG a vendes i albarans quan no es troba línia de matriu associada a les dimensions.	PR078	Canviar la dimensió del llarg a 3.	Columna preu per unitat secundària: 0 Columna preu unitari: 0 Columna subtotal: 0

Com a gestor vull triar el preu per unitat secundària dels PG a vendes i factures

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR091	Modificar el preu unitari a partir del preu per unitat secundària a vendes i factures amb quantitat establida a 1.	PR001	Introduir el producte 'Biga' a una venda o factura. Establir 1 unitat. Establir les dimensions llarg=5, ample=10 i alt=15. Establir el preu per unitat secundària a 10.	Columna preu unitari: 0,75 Columna subtotal: 0,75
PR092	Modificar el preu unitari a partir del preu per unitat secundària a vendes i factures amb quantitat > 1.	PR001	Introduir el producte 'Biga' a una venda o factura. Establir 2 unitats. Establir les dimensions llarg=5, ample=10 i alt=15. Establir el preu per unitat secundària a 10.	Columna preu unitari: 0,75 Columna subtotal: 1,50
PR093	Modificar el preu unitari a partir del preu per unitat secundària a vendes i factures amb unitat de mesura diferent de l'establida per defecte al producte i quantitat > 1.	PR001	Introduir el producte 'Biga' a una venda o factura. Establir 2 dotzenes. Establir les dimensions llarg=5, ample=10 i alt=15. Establir el preu per unitat secundària a 10.	Columna preu unitari: 0,75 Columna subtotal: 18

Com a gestor vull triar el preu unitari dels PG a vendes i factures i calcular el preu per unitat secundària

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR094	Modificar el preu per unitat secundària a partir del preu unitari a vendes i factures amb quantitat establida a 1.	PR001	Introduir el producte 'Biga' a una venda o factura. Establir 1 unitat. Establir les dimensions llarg=5, ample=10 i alt=15. Establir el preu unitari a 0,75.	Columna preu per unitat secundària: 10 Columna subtotal: 0,75

Com a gestor vull indicar a vendes/albarans/factures que la fusta emprada és PEFC

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR095	Indicar a una venda, albarà, factura que la fusta emprada és PEFC.		Crear una nova venda, albarà o factura i triar l'opció de fusta PEFC.	La venda, albarà o factura s'emmagatzema amb la informació sobre si s'utilitza fusta PEFC.

Com a gestor vull filtrar vendes/albarans/factures per si empren o no fusta PEFC

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR096	Filtrar vendes, factures i albarans que utilitzen fusta PEFC.	Crear 2 vendes, 2 albarans i 2 factures que utilitzen fusta PEFC i 2 vendes, 2 albarans i 2 factures que no empren fusta PEFC.	Seleccionar des de la vista llista de vendes, albarans i factures l'opció Filtres > Afegir filtre personalitzat, triar el camp que fa referència a la fusta PEFC i dir que el valor ha de ser vertader.	S'han de mostrar només les 2 vendes, albarans o factures que empren fusta PEFC.
PR097	Filtrar vendes, factures i albarans que no utilitzen fusta PEFC.	Crear 2 vendes, 2 albarans i 2 factures que utilitzen fusta PEFC i 2 vendes, 2 albarans i 2 factures que no empren fusta PEFC.	Seleccionar des de la vista llista de vendes, albarans i factures l'opció Filtres > Afegir filtre personalitzat, triar el camp que fa referència a la fusta PEFC i dir que el valor ha de ser fals.	S'han de mostrar només les 2 vendes, albarans o factures que no empren fusta PEFC.

Com a gestor vull configurar productes com a tractaments

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR098	Crear un tractament.		Crear un nou producte 'Vernís' de tipologia 'Servei' i seleccionar que es tracta d'un tractament.	S'afegeix a la base de dades un nou 'Servei' amb l'opció que indica que és un tractament establida a vertader.
PR099	Crear tractaments que funcionen com a productes Servei d'Odoo.		Crear un nou producte de tipologia diferent de 'Servei'.	L'opció per seleccionar que és un tractament no està disponible.

Com a gestor vull triar quins tractaments accepta un producte

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR100	Seleccionar com a tractaments només productes definits com a tractaments.	PR98 Crear un segon tractament 'Pintura'	Configurar i consultar la llista de possibles tractaments per a un producte des de la fitxa de creació o modificació del producte.	Només han d'aparèixer com a possibles tractaments els productes 'Pintura' i 'Vernís'
PR101	Atribuir possibles tractaments als productes.	PR001 PR098 PR100	Configurar el producte 'Biga' i seleccionar com a tractaments possibles 'Pintura' i Vernís'.	S'emmagatzema que el producte 'Biga' accepta els tractaments 'Pintura' i 'Vernís'.

Com a gestor vull aplicar tractaments a un producte des d'una venda seleccionant el preu del tractament

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR102	Aplicar tractaments a productes de les vendes.	PR101	Seleccionar el producte 'Biga' a una venda. Assignar-li a la línia del producte 'Biga' preu unitari = 10. Aplicar a 'Biga' el tractament 'Vernís' amb preu 1. Aplicar a 'Biga' el tractament 'Pintura' amb preu 1,5.	Apareixen a la venda 3 línies: una de 'Biga', una de 'Pintura' i una de 'Vernís'. Les línies de 'Pintura' i 'Vernís' apareixen d'un color diferent perquè són tractaments. Preu total de la venda: 12,5
PR103	Eliminar tractaments d'un producte en una venda.	PR102	S'elimina el tractament 'Vernís'.	Desapareix la línia 'Vernís' de la venda. Línies disponibles: 'Pintura' i 'Biga' Preu total de la venda: 11,5
PR104	Eliminar tractaments d'un producte a una venda quan s'elimina el producte.	PR103	S'elimina la línia del producte 'Biga'.	Desapareix la línia del producte 'Biga' i totes les línies dels productes referents als seus tractaments (la venda es queda buida). Preu total de la venda: 0

Com a gestor vull que els tractaments aplicats a un producte en una venda es traslladen als albarans i factures

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR105	Traspassar els tractaments de les vendes als albarans.	Crear una venda com la de la prova PR102.	Confirmar la venda.	Es genera un albarà. A la línia de l'albarà del producte 'Biga' s'indica que s'apliquen els tractaments 'Pintura' i 'Vernís'. A l'albarà només apareix una línia amb el producte 'Biga' (no apareix una línia corresponent a cada tractament).
PR106	Traspassar els tractaments de les vendes a les factures.	PR105	Se selecciona l'opció de crear una factura a partir d'una venda.	Es genera una factura. La factura conté tres línies: una del producte 'Biga', una del tractament 'Pintura' i una tercera del tractament 'Vernís'. Les línies dels tractaments apareixen d'un color diferent.

Com a gestor vull que als documents de vendes/albarans/factures es mostre la informació sobre els nous tipus de productes

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR107	Visualitzar la informació dels PG, PVM i PVC als PDF de vendes, albarans i factures.	PR001, PR002, PR003, PR004, PR005, PR006 Crear una venda amb els productes Biga, Tauler, Llistó, Barilla, Serradures i Troncs i generar un albarà i una factura a partir d'ella.	Generar el PDF de la venda/albarà/factura.	Columnes de dimensions del PDF: Només omplides a les línies dels PG. Columna d'unitat secundària: Omplida a les línies dels PG i PVM.
PR108	Ocultar columnes no emprades als PDF de vendes, factures i albarans quan els productes presents no són especials.	Crear una venda sense productes especials i generar un albarà i una factura a partir d'ella.	Generar el PDF de la venda/albarà/factura.	No apareixen les columnes de dimensions ni d'unitats secundàries als documents.

Com a gestor vull que als documents de vendes/factures es mostre el preu per unitat secundària dels PG

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR109	Visualitzar el preu per unitat secundària dels PG als PDF de vendes i factures.	PR001, PR002, PR003, PR004, PR005 Crear una venda amb els productes 'Biga', 'Tauler', 'Llistó', 'Barilla' i 'Serradures'. Generar una factura a partir de la venda.	Generar el PDF de la venda/factura	Apareix al document una columna del preu per unitat secundària. La columna de preu per unitat secundària està omplida a les línies de 'Biga', 'Tauler', 'Llistó' i 'Barilla'. La columna de preu per unitat secundària està buida a la línia de 'Serradures'.
PR110	Ocultar columnes no emprades als PDF de vendes, factures i albarans quan els productes presents no són especials.	Crear una venda sense productes especials i generar un albarà i una factura a partir d'ella.	Generar el PDF de la venda/albarà/factura.	No apareix la columna de preu per unitat secundària.

Com a gestor vull distingir els tractaments als documents de vendes

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR111	Visualitzar els tractaments als PDF de vendes.	PR105	Generar el PDF de la venda	Les línies referents als tractaments del producte 'Biga' ('Pintura' i 'Vernís') apareixen amb un format diferent.

Com a gestor vull saber quins tractaments s'apliquen a un producte amb el document d'albarà

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR112	Visualitzar els tractaments als PDF d'albarans.	PR105	Generar el PDF de l'albarà.	Apareix una columna al document per indicar els tractaments aplicats a un producte. La línia del producte 'Biga' conté, a la columna de tractaments, referència als tractaments aplicats ('Pintura' i 'Vernís').

Com a gestor vull distingir els tractaments dels productes tractats als documents de factura

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR113	Visualitzar els tractaments als PDF de factura.	PR106	Generar el PDF de la factura.	Les línies referents als tractaments del producte 'Biga' ('Pintura' i 'Vernís') apareixen amb un format diferent.

Com a gestor vull saber amb un document de venda/albarà/factura si la fusta emprada és PEFC

ID	Objectiu	Precondicions	Entrades	Resultats esperats
PR114	Visualitzar als PDF de vendes, albarans i factures si la fusta emprada és PEFC.	Crear una venda, albarà i factura que empra fusta PEFC.	Generar el PDF de la factura, albarà o factura.	El PDF ha d'indicar que la fusta emprada és de tipus PEFC.
PR115	Visualitzar els PDF de vendes, albarans i factures normals si la fusta emprada no és PEFC.	Crear una venda, albarà i factura que no empra fusta PEFC.	Generar el PDF de la factura, albarà o factura.	El PDF no ha d'indicar que la fusta emprada és de tipus PEFC.

5. Disseny

Una vegada establits els requisits es passa a la fase de disseny, guiada a través d'una sèrie de proves d'acceptació definides a l'apartat anterior. La secció s'inicia introduint l'arquitectura d'Odoo, aspecte clau per entendre com funciona l'ERP. A continuació, es parla de com treballen els mòduls a Odoo com a unitats de funcionalitat. La distribució modular del *software* en el seu conjunt fa que la divisió d'aquest projecte en mòduls que agrupen una sèrie de funcionalitats sembla la millor opció. Una vegada establits els mòduls en què es dividirà el desenvolupament, es passa a explicar de quina manera caldrà adaptar els models base a través de l'herència així com els nous models que s'hauran de crear per donar suport als requisits recollits.

5.1. Arquitectura d'Odoo

Odoo utilitza una arquitectura multicapa en tres capes: presentació, negoci i dades¹².

- La **capa de dades** emmagatzema les dades a una base de dades i les passa a la capa de negoci perquè realitzi les operacions. El sistema de gestió de bases de dades emprat és PostgreSQL.
- La **capa de negoci** s'encarrega de la lògica, els càlculs i les validacions. Aquesta capa és l'única que ha de tindre accés a la capa de dades per tal de mantindre tant la seguretat com la consistència de la informació. Aquest accés es realitza a través d'un motor de mapatge objecte-relacional (ORM). Tal com es defineix al web de l'empresa implantadora d'Odoo cyborsys.com, l'ORM “actua de pont entre el llenguatge de programació i la base de dades”. Així, “ajuda a executar ordres SQL sense escriure-les explícitament”¹³. D'aquesta manera, l'ORM d'Odoo proporciona una API que permet crear mòduls que interactuen amb la base de dades seguint els principis de la programació orientada a objectes i ofereix elements que permeten crear models, realitzar herències o utilitzar decoradors entre d'altres.
- La **capa de presentació** mostra les dades i interactua amb l'usuari, a més de realitzar validacions al costat del client.

L'existència d'una arquitectura definida té implicacions a l'hora de desenvolupar nous mòduls per a Odoo. Per exemple, comptar amb l'ORM fa que no siga necessari (almenys *a priori*) ocupar-se de la gestió de les dades a la base de dades, ja que les operacions CRUD es gestionen a través de codi Python mitjançant l'API proporcionada. També, l'existència d'aquestes interfícies implica que aspectes com la definició de nous models o la creació o supressió de registres s'hagen de realitzar seguint uns patrons ja determinats.

¹² REIS, Daniel. *Odoo 12 Development Essentials*. Packt Publishing, 2018. ISBN 1789532477

¹³ *ORM (Object Relational Mapping) in Odoo* [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.cyborsys.com/blog/orm-in-odoo>

5.2. Mòduls a Odoo

Odoo es compon de tota una sèrie de mòduls instal·lables. Aquests mòduls poden tant afegir noves funcionalitats com modificar el comportament d'altres lògiques de negoci. Els mòduls d'Odoo estan formats per una sèrie d'elements, que s'hauran d'afegir o no segons les característiques de cada desenvolupament concret. Aquests elements són els següents:

- **Objectes de negoci**, en forma de classes Python.
- **Objectes de vista**. Fitxers XML que defineixen la interfície gràfica.
- **Fitxers data**. Poden aparèixer en XML o CSV i serveixen per a definir dades dels models com, per exemple, informes, regles de seguretat o instàncies predefinides d'alguns objectes.
- **Controladors web**. S'encarreguen de donar resposta a les peticions del navegador web.
- **Elements web estàtics**, com imatges o fitxers CSS.

Per al present projecte serà suficient amb la generació de mòduls que continguin tres d'aquests elements: objectes de negoci, objectes de vista i fitxers data.

Tots els mòduls necessiten un fitxer `__manifest__.py`, encarregat de declarar el mòdul. Els fitxers `__manifest__.py` contenen, entre altres coses, el nom del mòdul, un resum del seu funcionament, dependències amb altres mòduls i la relació d'aquells fitxers que cal carregar (com els relacionats amb les vistes).

5.2.1. Objectes de negoci

Els objectes de negoci es declaren com a classes Python que estenen el model 'Model', la qual cosa fa que tots aquests objectes s'integren automàticament en un sistema de persistència¹⁴. Així, i com ja s'ha explicat a l'apartat sobre l'arquitectura d'Odoo, no serà necessari escriure la gran majoria de declaracions SQL.

Els models poden tindre definits una sèrie d'atributs. L'únic d'ells requerit és l'atribut `_name`, que ha de ser únic i que serveix per a identificar-lo. Altres, com `_description`, és també comú a tots els models encara que no obligatori.

Pel que fa als atributs de les classes, es defineixen de la següent manera:

```
attribute_name = fields.field_type()
```

¹⁴ *Building a Module* [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.odoo.com/documentation/13.0/howtos/backend.html>

Als atributs se'ls pot passar tota una sèrie de paràmetres. Alguns d'ells són comuns a tots els tipus de camps (com, per exemple, el paràmetre 'required' per indicar si l'atribut és obligatori o el paràmetre 'string' per definir la cadena de caràcters que apareixerà quan es mostre el camp en una vista) i d'altres són específics als camps d'un tipus determinat (com, per exemple, el paràmetre comodel_name per als camps de tipus Many2one, One2many o Many2many).

Per exemple, si es vol definir un atribut anomenat secondary_name que siga de tipus Char i que per defecte tinga valor 'Undefined', es fa de la següent manera:

```
secondary_name = fields.Char(  
    string='Secondary Name',  
    default='Undefined'  
)
```

A tots els models existeix un camp 'id', creat automàticament. Aquest emmagatzema un valor enter que identifica una instància d'una classe determinada.

5.2.1.1. Camps relacionals

Un model pot contindre camps que el relacionen amb instàncies d'ell mateix o d'un altre model. Aquestes relacions s'especifiquen en la definició de paràmetres dels camps i poden ser de tres tipus.

- **Many2one.** Una instància d'un model es relaciona amb una instància d'un altre model.

```
class MyClass(models.Model):  
    _name = "my.class"  
    other_class_id = fields.Many2one(  
        comodel_name="other.class"  
    )
```



Diagrama 3

Si la relació entre una instància de MyClass i una instància d'OtherClass és obligatòria, la definició es realitza de la següent manera:

```
class MyClass(models.Model):  
    _name = "my.class"  
    other_class_id = fields.Many2one(  
        comodel_name="other.class",  
        required=True  
    )
```



Diagrama 4

- **One2many.** Relació inversa al Many2one: una instància d'un model es relaciona amb diverses instàncies d'un altre model. Si en un model es defineix un camp One2many, s'ha de definir, necessàriament, un camp Many2one al model al qual fa referència.

```

class MyClass(model.Models):
    _name= "my.class"
    other_class_ids = fields.One2many(
        comodel_name="other.class",
        inverse_name="my_class_id"
    )
class OtherClass(model.Models):
    _name="other.class"
    my_class_id = fields.Many2one(
        comodel_name="my.class",
    )
  
```

El paràmetre `inverse_name` de l'atribut `One2many` indica el nom del camp `Many2one` del model amb què es relaciona.



Diagrama 5

- **Many2many.** Relació bidireccional en què qualsevol instància d'una classe pot estar relacionada amb qualsevol número d'instàncies d'una altra classe.

```

class MyClass(model.Models):
    _name = "my.class"
    other_class_ids = fields.Many2many(
        comodel_name="other.class"
    )
  
```



Diagrama 6

5.2.1.2. Camps computats

De vegades, el valor d'un camp s'ha de calcular a partir del resultat d'una funció. Per exemple, es podria tindre un camp `id_name` de tipus `Char` al model `res.partner` (que fa referència als contactes) que fos el resultat de concatenar l'identificador d'un contacte amb el qual la classe té una relació de tipus `Many2one` i el seu nom amb el format "ID/nom". Aquest cas es definiria de la següent manera:

```
id_name = fields.Char(
    compute = '_compute_id_name'
)
partner_id = fields.Many2one(
    comodel_name='res.partner'
)
def _compute_id_name(self):
    for record in self:
        record.id_name = "%s/%s" % (record.partner_id.id, record.partner_id.name)
```

En el cas anterior, cada vegada que es consulta l'atribut `id_name` s'executa la funció `_compute_id_name` per tal d'establir un valor. Açò vol dir que el valor no està emmagatzemat a la base de dades, la qual cosa té algunes conseqüències. Per exemple, si a una vista que mostra una llista d'instàncies apareix el camp `id_name`, cada vegada que s'accedeix a aquesta vista s'ha de calcular el valor d'`id_name` de totes les instàncies que hi apareixen, i això pot tindre un cost computacional important. A més a més, el funcionament base d'Odoo no permet filtrar instàncies en vistes de tipus llista a partir de camps computats no emmagatzemats en base de dades.

Per això, Odoo permet indicar si es vol emmagatzemar el valor d'un camp computat a la base de dades de la següent manera:

```
id_name = fields.Char(
    compute = '_compute_id_name',
    store=True
)
```

Ara es planteja una qüestió: què passa si canvia el valor del camp `partner_id`? Evidentment, caldria també actualitzar el valor d'`id_name` però, com que aquest està a la base de dades i ara ja no es recalcula cada vegada que es consulta, s'haurà d'indicar que la funció de càlcul s'ha d'executar cada vegada que canvia el valor del camp `partner_id`. Odoo ofereix una sèrie de decoradors de funcions i, entre ells, es troba el decorador 'depends'. La funció hauria de quedar de la següent manera:

```

@api.depends('partner_id')
def _compute_id_name(self):
    for record in self:
        record.id_name = "%s/%s" % (record.partner_id.id, record.partner_id.name)

```

Amb el paràmetre 'store' amb valor vertader al camp id_name s'indica que es vol emmagatzemar el valor del camp computat a la base de dades. Per aconseguir que aquest valor es recalculi cada vegada que canvia el valor del camp 'partner_id' s'utilitza el decorador 'depends', per assegurar que el valor d'id_name es troba sempre actualitzat.

Si el decorador 'depends' s'empra per a calcular els valors dels camps computats cada vegada que es produeixen operacions d'escriptura que afecten l'atribut o atributs indicats, existeix un altre decorador que activa la funció a la qual acompanya cada vegada que hi ha un canvi a l'atribut que porta entre parèntesis. Es tracta del decorador 'onchange' i també s'emprarà en el present desenvolupament. Es veuran exemples d'ús d'aquest decorador quan s'expliquen els mètodes que cal implementar en cadascun dels mòduls.

5.2.1.3. Restriccions

Un altre dels decoradors oferits per Odoo és 'constrains', emprat per establir restriccions. Per exemple: es disposa d'un camp de tipus Integer que només accepta valors positius. Aquest control es pot realitzar amb una restricció, que quedaria de la següent manera:

```

int_value = fields.Integer()

@api.constrains('int_value')
def _check_int_value(self):
    for record in self:
        if record.int_value < 0:
            raise ValidationError('int_value needs to be a positive integer!')

```

5.2.1.4. Seguretat

De manera nativa, Odoo ja s'encarrega de les qüestions de seguretat dels models base. Quan es desenvolupen nous models, però, cal definir qui en té accés.

En primer lloc, cal definir un grup d'usuaris del sistema, encara que també es poden aprofitar grups d'usuaris ja existents. Una vegada s'ha escollit un grup, s'indiquen quins permisos tindran els seus usuaris amb respecte un model concret. Aquesta informació s'especifica a un fitxer CSV. Per exemple, d'una banda, al grup 'general_administration' se'l vol donar accés total al model

'generic.product.pricelist'. D'altra, al grup 'employee' només se'l vol donar permisos de lectura. El fitxer quedaria de la següent manera:

```
id,name,model_id/id,group_id/id,perm_read,perm_write,perm_create,perm_unlink
generic_pricelist_edit,generic.pricelist.edit,generic_product_pricelist,general_administration
,1,1,1,1
generic_pricelist_read,generic.pricelist.read,generic_product_pricelist,employee ,1,0,0,0
```

5.2.1.5. Wizards

A Odoo es poden definir assistents, sessions interactives amb l'usuari en una finestra de diàleg¹⁵. Per definir un Wizard, el model ha d'heretar de la classe TransientModel en lloc de la classe Model. Els assistents són eliminats una vegada han finalitzat i no accepten la definició de permisos.

5.2.1.6. Herències

Quan es desenvolupa un nou mòdul, de vegades és necessari crear noves classes, però d'altres s'haurà de recórrer a l'herència. Odoo facilita dos tipus d'herències entre models: l'herència tradicional i l'herència per delegació¹⁶.

- **Herència tradicional.** S'empra per a modificar el comportament d'un model existent. A través d'aquest tipus d'herència es poden afegir nous camps, reescriure la definició dels camps dels models, afegir restriccions i reescriure i afegir mètodes. Aquesta tipologia es pot dividir en dos subtipus:
 - **Herència de classe.** S'afegeixen noves característiques a la classe. Les dades s'emmagatzemen en la mateixa taula que la superclasse i les vistes ja existents són compatibles. Aquest és el tipus d'herència que s'utilitza en aquest projecte. La definició d'aquesta herència es realitza de la següent manera:

```
class ClassInheritance(models.Model):
    _inherit = "class.parent"
```

El nom de ClassInheritance (atribut `_name`) serà el mateix que el de la superclasse.

- **Herència de prototip.** Amb aquesta herència es copien les característiques de la superclasse, però s'utilitza una nova taula per a emmagatzemar les dades i les vistes

¹⁵ *Building a Module* [en línia]. [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.odoo.com/documentation/13.0/howtos/backend.html>

¹⁶ *Building a Module* [en línia]. [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.odoo.com/documentation/13.0/howtos/backend.html>

existents són ignorades (per tant, caldrà desenvolupar de noves). La definició es realitza de la següent manera:

```
class ClassInheritance(models.Model):
    _name="class.inheritance"
    _inherit="class.parent"
```

Aquesta herència s'aconsegueix establint un nom diferent de la subclasse.

- **Herència per delegació.** Aquest tipus permet l'herència múltiple. Les instàncies de la nova classe contenen encastades una instància de la superclasse o superclasses amb les dades sincronitzades. Les dades de les instàncies de la nova classe i de la superclasse s'emmagatzemen en taules diferents. Aquest és el tipus d'herència que s'utilitza, per exemple, amb les classes de plantilla de producte i variant de producte.

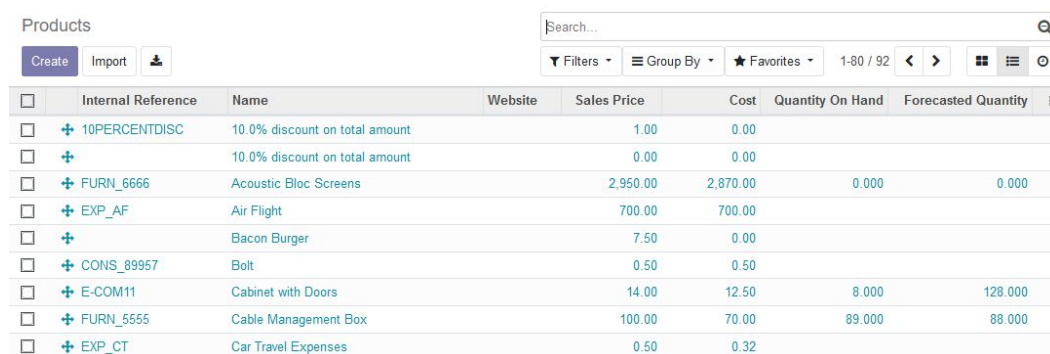
```
class ProductProduct(models.Model):
    _name = "product.product."
    _inherits = {'product.template': 'product_template_id'}
```

5.2.2. Vistes

Les vistes defineixen com es mostren les instàncies dels models. De vegades, quan es crea un nou model, és necessari definir vistes completament noves. D'altres, per exemple quan s'afegeixen nous atributs a un model, serà necessari utilitzar l'herència i ampliació d'una altra vista.

Existeixen diferents tipus de vistes. Les principals són les següents:

- **Vistes tree.** Mostra les instàncies tabularment.



<input type="checkbox"/>	Internal Reference	Name	Website	Sales Price	Cost	Quantity On Hand	Forecasted Quantity	!
<input type="checkbox"/>	+ 10PERCENTDISC	10.0% discount on total amount		1.00	0.00			
<input type="checkbox"/>	+	10.0% discount on total amount		0.00	0.00			
<input type="checkbox"/>	+ FURN_6666	Acoustic Bloc Screens		2.950.00	2.870.00	0.000	0.000	
<input type="checkbox"/>	+ EXP_AF	Air Flight		700.00	700.00			
<input type="checkbox"/>	+	Bacon Burger		7.50	0.00			
<input type="checkbox"/>	+ CONS_89957	Bolt		0.50	0.50			
<input type="checkbox"/>	+ E-COM11	Cabinet with Doors		14.00	12.50	8.000	128.000	
<input type="checkbox"/>	+ FURN_5555	Cable Management Box		100.00	70.00	89.000	88.000	
<input type="checkbox"/>	+ EXP_CT	Car Travel Expenses		0.50	0.32			

Captura 13

De manera bàsica, es declaren de la següent manera:

```
<tree string="Nom de la vista">
    <field name="nom_camp_mostrar" />
</tree>
```


- **Vistes formulari.** S'utilitzen per crear i editar les instàncies.

The screenshot shows the SAP Product Master Data interface for 'Acoustic Bloc Screens' (Product Code: FURN_6666). The interface includes a header with 'Products / [FURN_6666] Acoustic Bloc Screens', buttons for 'Edit', 'Create', 'Print', and 'Action', and a page indicator '1 / 80'. Below the header, there are tabs for 'Update Quantity' and 'Replenish'. A toolbar contains icons for 'Go to Website', '0.000 Units On Hand', '0.000 Units Forecasted', 'Traceability', '0 Reordering Ru...', '0 Bill of Materials', and 'Routes'. The main content area is titled 'Acoustic Bloc Screens' and features a small image of the product. It includes a list of checkboxes: 'Can be Sold' (checked), 'Can be Purchased' (checked), and 'Can be Expensed' (unchecked). Below this, there are tabs for 'General Information', 'Variants', 'Sales', 'eCommerce', 'Point of Sale', 'Purchase', and 'Inventory'. The 'General Information' tab is active, displaying a table with the following data:

Product Type	Storable Product	Sales Price	\$ 2,950.00
Product Category	All / Saleable / Office Furniture	Customer Taxes	
Internal Reference	FURN_6666	Cost	\$ 2,870.00 Update Cost
Barcode		Company	

Captura 14

De manera bàsica, es declaren de la següent manera:

```
<form string="Nom de la vista">
  <field name="nom_camp_mostrar" />
</form>
```

Els elements 'field' poden contindre tota una sèrie d'atributs per aconseguir, entre altres coses, ocultar els camps, mostrar-los només si es compleixen unes condicions o donar-los un format determinat.

5.2.2.1. Herències

De la mateixa manera que ocorre amb els models, les vistes es poden heretar. A continuació es mostra un exemple d'herència de vista.

```
<record id="nou_model_nom_vista_heretada" model="ir.ui.view">
  <field name="name">nou.model.nom.vista.heretada</field>
  <field name="model">nom.model</field>
  <field name="inherit_id" ref="modul_vista_pare.nom_vista_heretada"/>
  <field name="arch" type="xml">
    <xpath expr="//field[@name=nom_camp_vista_pare]" position="after">
      <field name="nom_nou_camp" string="Nou camp"/>
    </xpath>
  </field>
</record>
```

A la peça de codi anterior, a l'element field amb nom "inherit_id" s'indica quina és la vista que s'hereta a través del nom del mòdul en què se situa aquesta vista i l'id d'aquesta. L'element xpath s'empra per a introduir modificacions a la vista pare. En aquest cas, se cerca el camp "nom_camp_vista_pare" i s'indica que després d'aquest camp ("position="after") es vol afegir el camp nom_nou_camp. El string indica quina nomenclatura apareixerà a la vista per indicar aquest camp.

5.2.3. Templates

Els *templates* són fitxers en format xml que s'empren per a definir diferents tipus de plantilles, des de models de correu electrònic fins a informes en PDF. Al present desenvolupament els *templates* s'hauran de tindre en consideració en tant que cal mostrar als informes de vendes, albarans i factures la nova informació introduïda. És per això que serà necessari utilitzar l'herència, que funciona d'un mode molt similar a com ho fa a les vistes, per tal d'ampliar els *templates* dels informes PDF d'alguns dels models del sistema.

5.3. Distribució del projecte en mòduls

Una vegada presentat l'ús dels mòduls a Odoo i la seua composició bàsica, s'estableix una divisió de les funcionalitats del projecte global en diferents mòduls. La decisió de realitzar una divisió modular respon a diferents raons:

- Una **millor organització del projecte**. Açò contribuirà a millorar el manteniment futur.
- La possibilitat d'aconseguir que **certes funcionalitats estiguen disponibles aïlladament**. Encara que l'acoblament entre els mòduls és inevitable, ja que uns necessitaran característiques que aporten altres per poder funcionar, amb una divisió en mòduls es pot aconseguir que, almenys, alguns d'ells es puguin utilitzar sense que totes les funcionalitats estiguen instal·lades.

Així, basant-se en la separació de funcionalitats establida a l'apartat 4.2.2.7, es divideix el projecte global en 5 mòduls, mostrats al Diagrama 7.

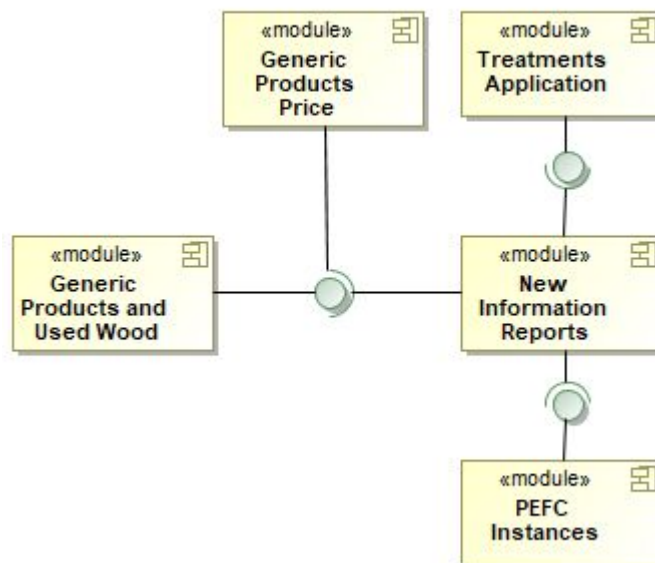


Diagrama 7

- **Generic Products and Used Wood.** Aquest mòdul defineix la tipologia de productes necessaris per al client i n'inclou les funcionalitats relacionades amb el càlcul de la fusta emprada i els m3, m2 i ml dels PG.
- **Generic Products Price.** Aquest mòdul inclou el preu per unitat secundària dels PG, les matrius de preus dels PG i la seua integració amb les vendes i les factures.
- **Treatments Applications.** Aquest mòdul inclou la possibilitat d'afegir tractaments als productes. No requereix el mòdul "Generic Products and Used Wood" perquè els tractaments es poden aplicar a qualsevol producte i no tenen un funcionament particular amb els productes especials.
- **PEFC instances.** Aquest mòdul simplement inclou la possibilitat d'indicar si les vendes, factures i albarans utilitzen fusta PEFC.
- **New Information Reports.** Inclou als informes de vendes, albarans i factures la nova informació aportada pels mòduls "Generic Products and Used Wood", "Generic Product Price", "Treatments Application" i "PEFC Instances".

5.4. Classes, vistes, mètodes, informes i data dels mòduls

Per al desenvolupament d'aquest projecte s'haurà d'utilitzar tant l'herència de models, vistes i *templates* com la definició de noves entitats. Aquest apartat es divideix, per tant, en dues parts. A la primera s'introdueixen els models d'Odoo base que s'han d'heretar. A continuació s'analitza cadascun dels mòduls en què es divideix el projecte per indicar quins atributs cal que tinguin als models utilitzats, quines operacions s'han d'implementar i com han de ser les vistes i els fitxers de dades. En el cas de les operacions, aquestes es defineixen dividides per funcionalitats i s'acompanyen també de les restriccions. Els mètodes estan documentats amb la seua signatura en Python segons els estàndards d'Odoo, una descripció de la seua utilitat i els decoradors que fan servir.

5.4.1. Models heretats

5.4.1.1. `product.template` i `product.product`

Les instàncies de `product.template` representen la plantilla d'un producte i les de `product.product` una variant de producte. Com s'ha comentat a l'apartat 4.2.1.1., `product.product` és una subclasse de `product.template`. Quan es crea una plantilla de producte, si no es defineixen variants, automàticament es crea una instància de `product.product` amb les dades sincronitzades.

Entre d'altres, trobem els següents atributs:

- **`product_type`**. Camp seleccionable per indicar si el producte és emmagatzemable, consumible o es tracta d'un servei.
- **`sale_ok`**. Camp de tipus Boolean que indica si el producte es pot vendre.
- **`purchase_ok`**. Camp de tipus Boolean que indica si el producte es pot comprar.
- **`can_be_expensed`**. Camp de tipus Boolean que indica si el producte és una despesa.
- **`list_price`**. Camp de tipus Float amb el preu de venda.
- **`standard_price`**. Camp de tipus Float amb el preu de compra.

5.4.1.2. sale.order i sale.order.line

Els models sale.order i sale.order.line instancien vendes i línies de vendes respectivament.

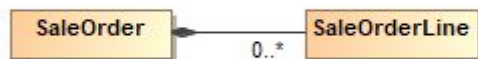


Diagrama 8

El primer d'ells conté, entre d'altres, els següents atributs:

- **partner_id**. Camp Many2one que indica el client de la venda.
- **pricelist_id**. Camp Many2one que indica la tarifa aplicada a la venda. S'actualitza automàticament amb la *pricelist* assignada al client.
- **amount_total**. Camp de tipus Float amb l'import total de la venda.

A sales.order.line es troben els següents atributs:

- **product_id**. Camp Many2one cap a una variant de producte.
- **name**. Camp de text amb el nom complet del producte.
- **product_uom_qty**. Camp de tipus Float amb la quantitat de producte que s'adquireix.
- **product_uom**. Camp Many2one cap a una UoM.
- **price_unit**. Camp de tipus Float amb el preu del producte.
- **tax_id**. Camp de tipus Many2many amb la relació d'impostos aplicats.
- **price_subtotal**. Camp de tipus Float amb el preu de tota la quantitat introduïda a la línia abans de l'aplicació d'impostos.

5.4.1.3. stock.picking, stock.move i stock.move.line

El model stock.picking fa referència a l'albarà. Un albarà té relacionades un conjunt d'instàncies de stock.move, que representen un moviment en el stock des d'un magatzem, i un conjunt d'instàncies de stock.move.line, que fan referència a moviments de stock des d'una ubicació concreta. Així, un stock.move es relaciona amb un conjunt de stock.move.line.

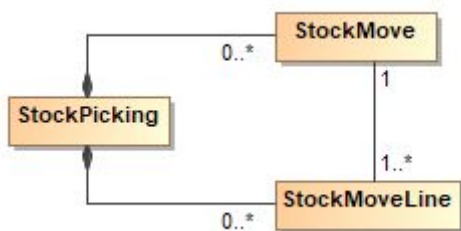


Diagrama 9

El model stock.picking conté, entre d'altres, els següents camps:

- **partner_id**. Camp Many2one cap a un contacte que representa l'adreça de lliurament dels productes.
- **scheduled_date**. Camp Daytime amb el dia i l'hora en què s'espera que s'haja lliurat el producte.
- **origin**. Camp de tipus char amb el nom de la venda de la qual procedeix l'albarà.

Al model stock.move es poden trobar els següents camps:

- **product_id**. Camp Many2one cap a la variant de producte.
- **product_uom_quantity**. Camp de tipus Float amb la quantitat de producte demandada.
- **reserved_availability**. Camp de tipus Float amb la quantitat del total reservada per a aquest albarà.
- **quantity_done**. Quantitat efectivament lliurada amb l'albarà.

Al model stock.move.line hi estan els següents camps:

- **product_id**. Camp Many2one cap a la variant de producte.
- **location_id**. Camp Many2one cap a la localització exacta del producte.
- **product_uom_qty**. Camp Float amb la quantitat reservada.
- **qty_done**. Camp Float amb la quantitat efectivament lliurada del producte amb l'albarà.

5.4.1.4. account.move i account.move.line

El model account.move fa referència a les factures i les instàncies d'account.move.line representen cadascuna de les línies de factura.

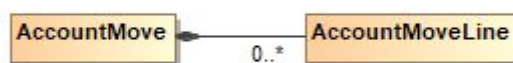


Diagrama 10

En account.move podem trobar els següents camps:

- **partner_id**. Camp Many2one cap a la instància de client.
- **pricelist_id**. Camp Many2one cap a la tarifa emprada. Aquest camp s'obté instal·lant un mòdul de la Odoo Community Association (a les vendes ja hi existeix per defecte).
- **amount_total**. Camp de tipus Float amb l'import total de la factura.

Alguns dels camps presents a account.move.line són els següents:

- **product_id**. Camp Many2one cap a una variant de producte.
- **name**. Camp de text amb el nom complet del producte.
- **quantity**. Camp de tipus Float amb la quantitat de producte que s'adquireix.
- **product_uom_id**. Camp Many2one cap a una UoM.

- **price_unit**. Camp de tipus Float amb el preu del producte.
- **tax_id**. Camp de tipus Many2many amb la relació d'impostos aplicats.
- **price_total**. Camp de tipus Float amb el preu de tota la quantitat introduïda a la línia amb l'aplicació d'impostos.

Una vegada explicats els models heretats principals per al projecte, es defineixen els atributs, operacions, vistes, dades i plantilles necessàries per a cadascun dels mòduls. Algunes de les operacions apareixen acompanyades de diagrames de seqüència, l'objectiu dels quals és **només mostrar de quina manera interactuen les diferents classes en realitzar algunes operacions i quina és la seqüència de crides dels diferents mètodes implicats**.

5.4.2. Generic Products and Used Wood

En aquest apartat s'exposen els models amb els atributs, els mètodes, restriccions i vistes presents al mòdul "Generic Products and Used Wood".

5.4.2.1. Atributs

Nous camps del model heretat product.template

Nom	Tipus	Funció
is_generic	Boolean	El producte és de tipus genèric.
generic_units	Selection	Quatre valors possibles: m3, m2, ml o stick.
use_default_dimensions	Boolean	El producte utilitza dimensions per defecte.
block_default_dimensions	Boolean	Les dimensions per defecte no es poden modificar a les vendes, albarans i factures.
item_lenth	Float	Llarg en metres per defecte.
width	Float	Ample en centímetres per defecte.
height	Float	Gruix en centímetres per defecte.
is_pvm	Boolean	El producte és de tipus PVM.
has_conversion_rate	Boolean	El producte és de tipus PVC.
conversion_rate	Float	Coefficient de conversió dels PVC.

Nous camps del model heretat product.product

Nom	Tipus	Funció
use_default_dimensions	Boolean	El producte utilitza dimensions per defecte.
block_default_dimensions	Boolean	Les dimensions per defecte no es poden modificar a les vendes, albarans i factures.
item_lenth	Float	Llarg en metres per defecte.
width	Float	Ample en centímetres per defecte.
height	Float	Gruix en centímetres per defecte.
conversion_rate	Float	Coeficient de conversió als PCV.

Nous camps del model heretat sale.order.line

Nom	Tipus	Funció
item_length	Float	Longitud del PG (en metres).
width	Float	Ample del PG (en centímetres).
height	Float	Gruix del PG (en centímetres).
m3	Float	Metres cúbics del PG.
m2	Float	Metres quadrats del PG.
ml	Float	Metres linials del PG.
m3_sold	Float	Fusta emprada pels PG, PVM o PVC.
is_m3	Boolean	El producte és PG de tipus m3.
is_m2	Boolean	El producte és PG de tipus m2.
is_ml	Boolean	El producte és PG de tipus ml.
is_stick	Boolean	El producte és PG de tipus pal.
show_generic_options	Boolean	Mostrar el llarg, l'ample i el gruix a la línia.
default_dimensions	Boolean	El producte té dimensions per defecte.
fixed_dimensions	Boolean	El producte té dimensions per defecte bloquejades.
show_volume	Boolean	Mostrar la fusta emprada.

has_conversion_rate	Boolean	El producte és de tipus PVC.
pvm_read_only	Boolean	Fals si el producte és PVM.

Nous camps del model heretat sale.order

Nom	Tipus	Funció
m3_sold	Float	Fusta total emprada pels PG, PVM i PVC.

Nous camps del model heretat stock.picking

Nom	Tipus	Funció
m3_sold	Float	Fusta total emprada pels PG, PVM i PVC.

Nous camps del model heretat stock.move

Nom	Tipus	Funció
item_length	Float	Longitud del PG (en metres).
width	Float	Ample del PG (en centímetres).
height	Float	Gruix del PG (en centímetres).
m3	Float	Metres cúbics del PG.
m2	Float	Metres quadrats del PG.
ml	Float	Metres lineals del PG.
m3_sold	Float	Fusta emprada pels PG, PVM o PVC.
is_m3	Boolean	El producte és PG de tipus m3.
is_m2	Boolean	El producte és PG de tipus m2.
is_ml	Boolean	El producte és PG de tipus ml.
is_stick	Boolean	El producte és PG de tipus pal.
show_generic_options	Boolean	Mostrar el llarg, l'ample i el gruix a la línia.
default_dimensions	Boolean	El producte té dimensions per defecte.

fixed_dimensions	Boolean	El producte té dimensions per defecte bloquejades.
show_volume	Boolean	Mostrar la fusta emprada.
has_conversion_rate	Boolean	El producte és de tipus PVC.
pvm_read_only	Boolean	Fals si el producte és PVM.

Nous camps del model heretat stock.move.line

Nom	Tipus	Funció
item_length	Float	Longitud del PG (en metres).
width	Float	Ample del PG (en centímetres).
height	Float	Gruix del PG (en centímetres).
m3	Float	Metres cúbics del PG.
m2	Float	Metres quadrats del PG.
ml	Float	Metres lineals del PG.
m3_sold	Float	Fusta emprada pels PG, PVM o PVC.
is_m3	Boolean	El producte és PG de tipus m3.
is_m2	Boolean	El producte és PG de tipus m2.
is_ml	Boolean	El producte és PG de tipus ml.
is_stick	Boolean	El producte és PG de tipus pal.
show_generic_options	Boolean	Mostrar el llarg, l'ample i el gruix a la línia.
default_dimensions	Boolean	El producte té dimensions per defecte.
fixed_dimensions	Boolean	El producte té dimensions per defecte bloquejades.
show_volume	Boolean	Mostrar la fusta emprada.
has_conversion_rate	Boolean	El producte és de tipus PVC.
pvm_read_only	Boolean	Fals si el producte és PVM.

Nous camps del model heretat account.move.line

Nom	Tipus	Funció
item_length	Float	Longitud del PG (en metres).
width	Float	Ample del PG (en centímetres).
height	Float	Gruix del PG (en centímetres).
m3	Float	Metre cúbics del PG.
m2	Float	Metres quadrats del PG.
ml	Float	Metres lineals del PG.
m3_sold	Float	Fusta emprada pels PG, PVM o PVC.
is_m3	Boolean	El producte utilitza m3 com a unitat secundària.
is_m2	Boolean	El producte utilitza m2 com a unitat secundària.
is_ml	Boolean	El producte utilitza ml com a unitat secundària.
is_stick	Boolean	El producte és PG de tipus pal.
show_generic_options	Boolean	Mostrar el llarg, l'ample i el gruix a la línia.
default_dimensions	Boolean	El producte té dimensions per defecte.
fixed_dimensions	Boolean	El producte té dimensions per defecte bloquejades.
show_volume	Boolean	Mostrar la fusta emprada.
has_conversion_rate	Boolean	El producte és de tipus PVC.
pvm_read_only	Boolean	Fals si el producte és PVM.

Nous camps del model heretat account.move

Nom	Tipus	Funció
m3_sold	Float	Fusta total emprada pels PG, PVM i PVC.

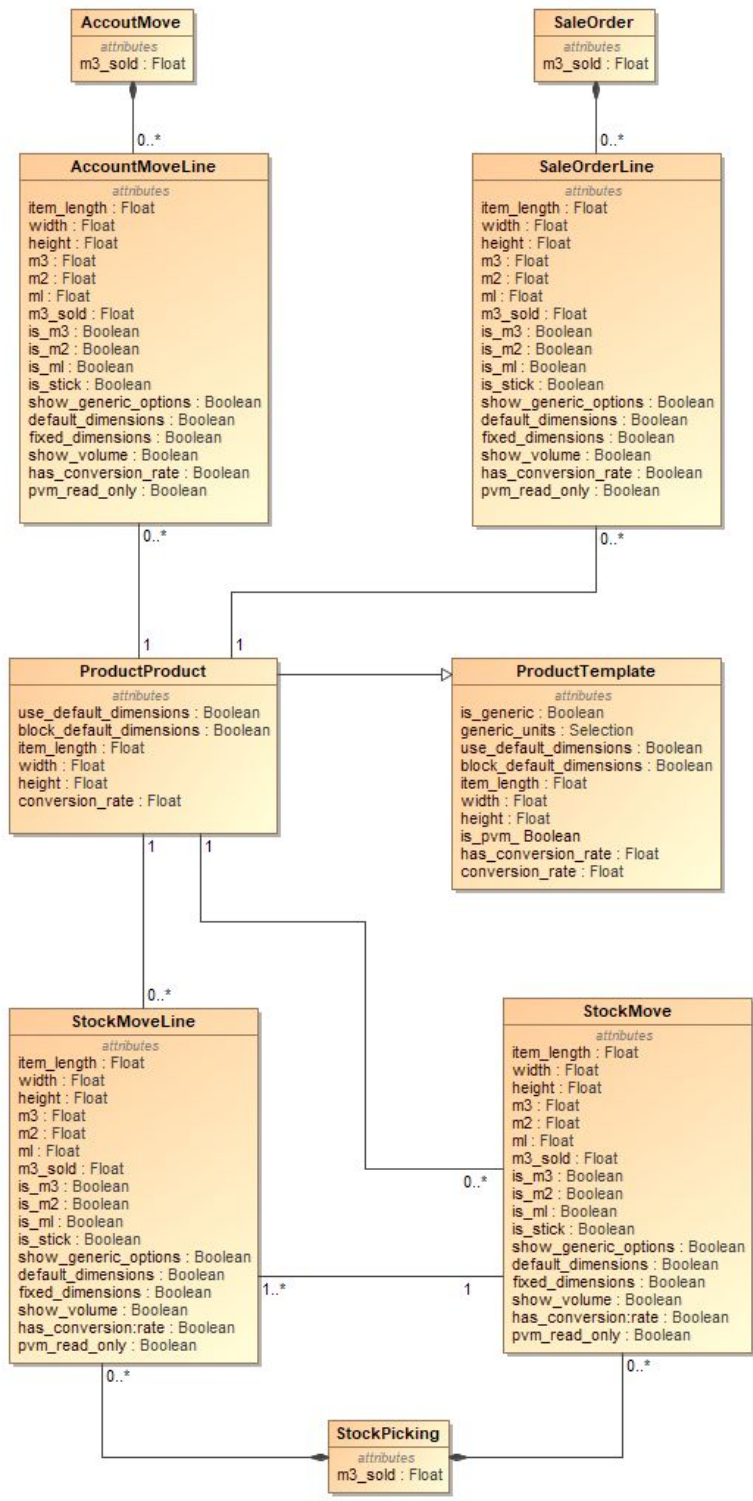


Diagrama 11

5.4.2.2. Operacions i restriccions

Definició del tipus de producte

Operacions

Els productes poden ser de tipus PG (amb els subtipus m3, m2, ml i pal), PVC o PVM. Si a una plantilla de producte se li assigna un d'aquests tipus, totes les seues variants haurien de conservar aquesta tipologia. Existeixen, així mateix, altres dades (com les dimensions per defecte) que són configurables a cadascuna de les variants.

Com s'ha explicat prèviament, quan es crea una plantilla de producte es genera, automàticament, una variant amb la informació sincronitzada, tant la dels atributs que defineix la *template* per totes les variants com la dels atributs configurables des de cada variant. Per exemple, si es defineix una *template* PG en m3 anomenada 'biga' i sense cap variant, es crearà una variant 'biga' que ha de mantindre la mateixa informació de la plantilla. Així, s'ha d'aconseguir que, si a la *template* es canvia, per exemple, el valor del llarg per defecte (configurable a cada variant en cas que la *template* en tinga més variants a més de la generada per defecte), es canvie a la variant única. També ha d'ocórrer al contrari: si es canvia el valor del llarg per defecte a la variant única, s'ha de canviar el llarg per defecte a la instància *template* de 'biga'. Per tal d'assolir aquest funcionament, se segueix un patró que altres dels atributs configurables a cada variant (per exemple `lst_price`) fan servir. S'explica a continuació amb l'exemple del camp `item_length`, present tant al model `product.template` com `product.product` (ja que es tracta d'un atribut configurable a cada variant).

A `product.product`, l'atribut es declara sense cap tipus de particularitat. És a dir:

```
item_length = fields.Float(  
    string="Length (m)"  
)
```

A `product.template`, en canvi, es defineix de la següent manera:

```
item_length = fields.Float(  
    string="Length (m)",  
    compute="_compute_length",  
    inverse="_set_length",  
    store=True  
)
```

Es tracta d'un camp computat emmagatzemat a la base de dades. Aquest fet fa que siga necessari que la funció que computa el valor de l'atribut tinga el decorador `@depends`. S'afegeix, així mateix, el nom d'una funció al paràmetre `inverse`. Amb açò s'aconsegueix, en primer lloc, que el camp deixe de ser només lectura i, en segon lloc, que s'haja d'adaptar el valor del qual depèn el càlcul de l'atribut en cas que es modifique el valor del camp manualment.

El codi de les funcions `_compute_length` i `_set_length` és el següent:

```

@api.depends('product_variant_ids', 'product_variant_ids.item_length')
def _compute_length(self):
    unique_variants = self.filtered(lambda temp: len(template.product_variant_ids) == 1)
    for template in unique_variants:
        template.item_length = template.product_variant_ids.item_length
    for template in (self - unique_variants):
        template.item_length = False

def _set_length(self):
    for template in self:
        template.product_variant_ids.item_length = template.item_length

```

La funció `_compute_length` porta el decorador `@depends` i s'indica que el valor de `item_length` de `product.template` s'ha de computar quan canvien els atributs `product_variant_ids` (relació One2Many amb les variants de producte de la *template*) o el valor de l'atribut `item_length` de les variants. A la funció, primer s'aplica el valor d'`item_length` de la variant a la *template* quan la *template* només té una variant (és a dir, la variant generada per defecte). Si la *template* té més d'una variant, el valor d'`item_length` a la plantilla es posa a `False`, ja que s'ha de gestionar des de cadascuna de les variants. És a dir: si es canvia el valor d'`item_length` a la variant única de biga, es dispara la funció `_compute_length`, que actualitza el valor d'`item_length` de la *template* perquè estiga sincronitzat amb el de la variant.

La funció `_set_length` es llançarà cada vegada que es canvia manualment el valor d'`item_length` des de la plantilla de producte per posar a totes les variants de la plantilla el valor introduït. És per això important configurar correctament la vista perquè, en cas que la *template* tinga més d'una variant, el valor configurable a cada variant (`item_length` en aquest cas) desaparega de la vista de la *template*, ja que, en cas que estiguera present, aquest valor es canviaria a totes les variants. En canvi, si el camp `item_length` només és visible a la vista de `product.template` quan la instància de la plantilla té una variant, és correcte canviar el valor d'`item_length` de totes les variants quan es canvie manualment des de la vista de la *template*, ja que només té una variant i els seus valors han d'estar sincronitzats.

El Diagrama 12 mostra com es calcula i trasllada el valor d'`item_length` quan es canvia manualment des de la variant única de producte.

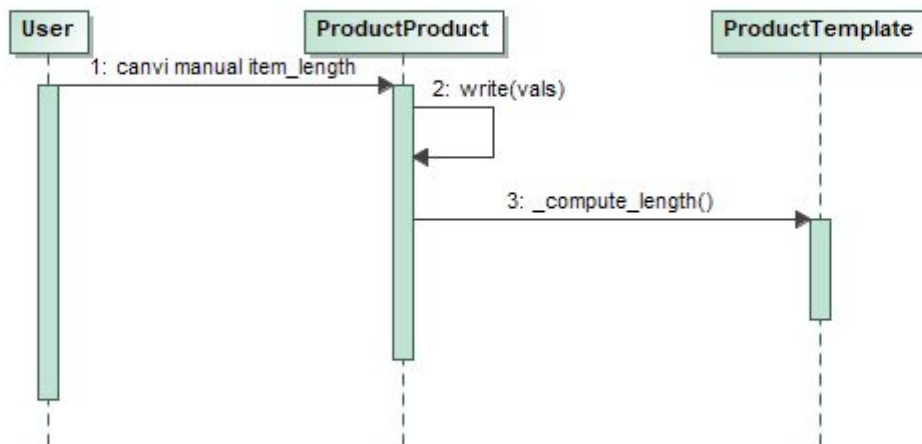


Diagrama 12

El Diagrama 13 mostra com, en canviar manualment el valor d'item_length a la plantilla d'un producte amb només una variant, aquest valor es replica a la variant.

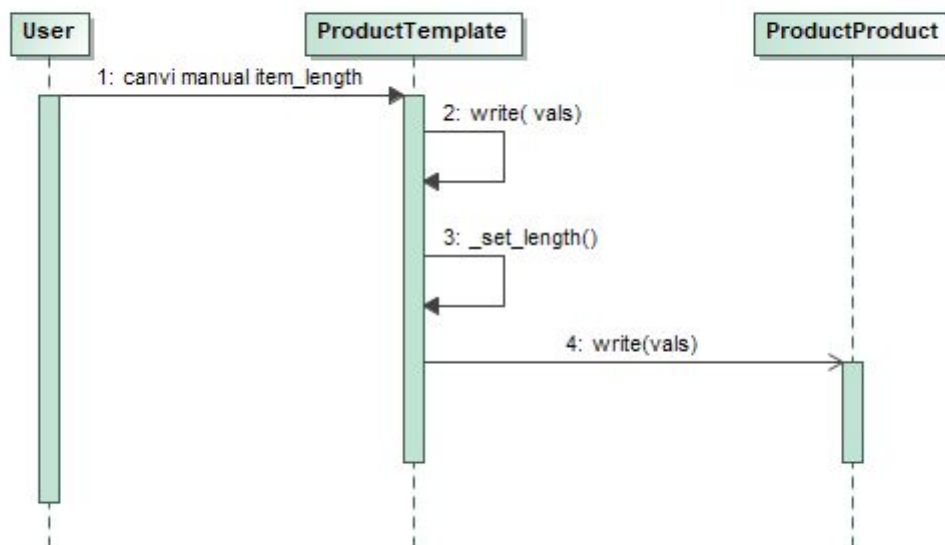


Diagrama 13

Seguint aquesta lògica, cal definir, per tant, un mètode `_compute` amb el decorador `depends` i un mètode `_set` per cada atribut de producte configurable des de cada variant per tal de mantindre aquests atributs sincronitzats entre variable i *template* quan la *template* només dispose d'una variant.

```

@api.depends('product_variant_ids', 'product_variant_ids.use_default_dimensions')
def _compute_use_default_dimensions(self):

def _set_use_default_dimensions(self):

@api.depends('product_variant_ids', 'product_variant_ids.block_default_dimensions')
def _compute_block_default_dimensions(self):

def _set_block_default_dimensions(self):

@api.depends('product_variant_ids', 'product_variant_ids.item_length')
def _compute_item_length(self):

def _set_item_length(self):

@api.depends('product_variant_ids', 'product_variant_ids.width')
def _compute_width(self):

def _set_width(self):

@api.depends('product_variant_ids', 'product_variant_ids.height')
def _compute_height(self):

def _set_height(self):

@api.depends('product_variant_ids', 'product_variant_ids.conversion_rate')
def _compute_conversion_rate(self):

def _set_conversion_rate(self):

```

Restriccions

product.template

```

@api.constrains('is_generic', 'is_pvm', 'has_conversion_rate')
def _check_unique_product_type(self)

```

Els productes només poden ser d'un tipus (PG, PVM, PVC o cap d'ells).

Indicar les dimensions dels PG en vendes, factures i albarans

Mètodes

sale.order.line, account.move.line, stock.move, stock.move.line

```
@api.depends('product_id')
```

```
def _compute_show_generic_options(self)
```

Aquest mètode determina si la línia de venda, factura o albarà conté un producte de tipus PG i, en conseqüència, indica a la vista si s'han de mostrar les dimensions. Actualitza el valor de `show_generic_option` segons la comprovació anterior.

Restriccions

sale.order.line, account.move.line, stock.move, stock.move.line

```
@api.constrains('item_length', 'width', 'height')
```

```
def _check_positive_dimensions(self)
```

Les dimensions dels PG a les línies de venda, factura o albarà han de ser valors positius. Si es detecta una dimensió amb valor negatiu, es llança un missatge d'error i no es permet guardar la instància de venda, albarà o factura.

```
@api.constrains('width', 'height')
```

```
def _check_stick_dimensions(self)
```

Quan s'utilitzen PG de tipus pal a les línies de venda, factura o albarà, ample ha de ser més gran que gruix. Si es detecta que a una línia amb un PG de tipus pal el gruix és major que l'ample, es llança un missatge d'error i no es permet guardar la venda, albarà o factura.

Definició de dimensions per defecte

Restriccions

product.product, product.template

```
@api.constrains('item_length', 'width', 'height')
```

```
def _check_positive_dimensions(self)
```

Les dimensions per defecte han de ser valors positius. Si es detecta alguna dimensió amb valor negatiu, es mostra un missatge d'error i no es permet guardar la plantilla de producte o la variant del producte.

```
@api.constrains('width', 'height')
```

```
def _check_stick_dimensions(self)
```

En els PG de tipus pal, ample ha de ser més gran que gruix. Si es detecta que gruix és major que l'ample en un PG de tipus pal, es mostra un missatge d'error i no es permet guardar la plantilla de producte o la variant de producte.

Mostrar les dimensions per defecte en vendes, factures i albarans

Mètodes

sale.order.line, account.move.line, stock.move, stock.move.line

```
@api.onchange('product_id')
def show_default_dimensions(self):
```

Si el producte és PG i aquest té definides dimensions per defecte, es mostraran a la línia de venda, albarà o factura quan se seleccione aquest producte.

Bloquejar les dimensions per defecte en vendes, factures i albarans

Mètodes

sale.order.line, account.move.line, stock.move, stock.move.line

```
@api.depends('product_id')
def _compute_fixed_dimensions(self)
```

Si el producte de la línia té indicat que les dimensions per defecte han d'estar bloquejades es posarà el valor de `fixed_dimensions` a vertader per tal d'indicar a la vista que els camps de dimensions d'eixa línia són *readonly*.

Calcular els m3, m2 i ml dels PG

Mètodes

product.product

```
def calculate_secondary_units(self, item_length, width, height, quantity, uom)
```

Calcula els m3, m2 o ml segons les dimensions i quantitats introduïdes i el tipus de producte en qüestió. Retorna un valor doble: el valor numèric corresponent als m3, m2 o ml i un string amb el tipus d'unitat secundària segons el producte.

sale.order.line, account.move.line, stock.move, stock.move.line

```
@api.depends('product_id')
def _compute_is_m3(self)
```

Si el producte de la línia és PG de tipus m3, posa el valor de is_m3 a vertader.

```
@api.depends('product_id')
def _compute_is_m2(self)
```

Si el producte de la línia és PG de tipus m2, posa el valor de is_m2 a vertader.

```
@api.depends('product_id')
def _compute_is_ml(self)
```

Si el producte de la línia és PG de tipus ml, posa el valor de is_ml a vertader.

```
@api.depends('product_id')
def _compute_is_stick(self)
```

Si el producte de la línia és PG de tipus pal, posa el valor de is_stick a vertader.

```
@api.depends('product_id')
def _compute_has_conversion_rate(self)
```

Si el producte de la línia és PVC, es posa el valor has_conversion_rate a vertader.

```
@api.depends('product_id')
def _compute_pvm_read_only(self)
```

Si el producte de la línia és PVM, es posa el valor m3l_read_only a fals.

sale.order.line

```
@api.depends('item_length', 'width', 'height', 'product_uom_qty', 'product_uom',
'product_id')
def _compute_secondary_units(self)
```

account.move.line

```
@api.depends('product_id', 'item_length', 'width', 'height', 'quantity', 'product_uom_id')
def _compute_secondary_units(self)
```

stock.move

```
@api.depends('product_id', 'item_length', 'width', 'height', 'product_uom_qty',
'product_uom', 'product_id', 'quantity_done')
def _compute_secondary_units(self)
```

stock.move.line

```
@api.depends('product_id', 'item_length', 'width', 'height', 'product_uom_qty',  
'product_uom_id', 'product_id', 'qty_done')  
def _compute_secondary_units(self)
```

Criden a la funció de product.product que retorna el valor de la unitat secundària i el tipus d'unitat secundària. Aquestes funcions actualitzaran el valor dels m3, m2 o ml amb el valor obtingut i el col·locarà a l'atribut corresponent segons la unitat secundària rebuda. Al Diagrama 14 es mostra aquest funcionament en el cas de les línies de venda.

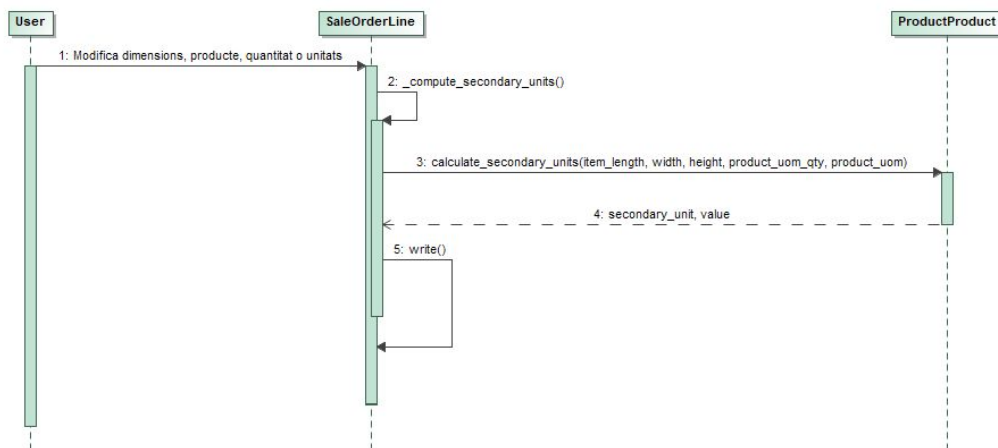


Diagrama 14

Calcular la fusta emprada en línies de vendes, factures i albarans

Mètodes

product.product

```
def calculate_used_wood(self, quantity, uom, item_length=False, width=False, height=False)
```

Calcula la fusta emprada segons el tipus de producte i les dimensions introduïdes. Els paràmetres del llarg, ample i alt tenen valor per defecte fals, ja que, en el cas dels productes PVC i PVM, també s'ha de calcular la fusta emprada, però no en tenen dimensions. Retorna un valor numèric amb la fusta emprada.

sale.order.line, account.move.line, stock.move, stock.move.line

```
@api.depends('product_id', 'm3', 'm2', 'ml')  
def _compute_m3_sold(self):
```

Calcula la fusta emprada a la línia de producte a través de la crida a la funció calculate_used_wood de product.product. La seqüència és similar a la del Diagrama 14.

Calcular la fusta emprada a vendes, albarans i factures

Operacions

sale.order, stock.picking, account.move

sale.order

```
@api.depends('order_line.m3_sold')
def _compute_m3_sold(self):
```

stock.picking

```
@api.depends('move_ids_without_package.m3_sold')
def _compute_m3_sold(self):
```

account.move

```
@api.depends('order_line.m3_sold')
def _compute_m3_sold(self):
```

Suma la fusta emprada per totes les línies de venda/factura/albarà i actualitza el valor del camp m3_sold amb aquest càlcul.

Passar la informació de dimensions de les línies de venda a les línies d'albarà

Operacions

sale.order.line

```
def _prepare_procurement_values(self):
```

Mètode heretat de la superclasse sale.order.line que es crida quan s'accepta una venda i es genera un albarà. Quan es crida al mètode de la superclasse, aquesta retorna un diccionari amb els valors que cal passar a les línies de l'albarà. A la funció de la subclasse cal actualitzar aquest diccionari amb els valors afegits amb aquest mòdul: dimensions del PG i valors dels metres cúbics, metres quadrats i metres lineals. El mètode heretat retorna aquest diccionari amb els valors actualitzats.

stock.rule

```
def _get_custom_move_fields(self)
```

Mètode heretat que recull el nom dels camps que cal passar als albarans que no formen part d'Odoo base. En primer lloc, cal cridar al mètode de la superclasse (stock.rule) per rebre una llista amb els camps addicionals i, a continuació, s'han d'afegir els camps addicionals

introduïts per aquest mòdul: les dimensions dels PG, els metres cúbics, els metres quadrats i els metres lineals. El mètode heretat retorna la llista amb el nom dels camps addicionals.

stock.move

```
def _prepare_move_line_vals(self):
```

Funció heretada que passa els valors de la línia de l'albarà a les línies detallades de l'albarà. En cridar a la funció de la superclasse, es rep un diccionari amb els valors que es passen de les línies de l'albarà a les línies detallades de l'albarà. Cal actualitzar aquest diccionari amb els valors introduïts als albarans amb aquest mòdul: les dimensions dels PG, els metres cúbics, els metres quadrats i els metres lineals.

Passar la informació de dimensions de les línies de venda a les línies de factura

Operacions

sale.order.line

```
def _prepare_invoice_line(self):
```

Mètode heretat de la superclasse `sale.order.line`. Quan crida al mètode de la superclasse, rep un diccionari amb els valors que cal passar a les línies de factura i cal actualitzar aquest diccionari amb els valors afegits amb aquest mòdul: dimensions del PG i valors dels metres cúbics, metres quadrats i metres lineals. El mètode heretat retorna aquest diccionari amb els valors actualitzats.

5.4.2.3. Vistes

product.product_template_form_view (heretada)

Aquesta vista heretada mostra la informació dels productes i és utilitzada tant per les plantilles com per les variants. El funcionament base d'Odoo d'aquesta vista, depenent si es tracta d'una plantilla o d'una variant, és el següent:

- Camps definits a la *template* comuns a totes les variants. Es mostren tant quan es consulta una plantilla com una variant.
- Camps particulars a cada variant de producte. Com s'ha explicat anteriorment, quan es crea una plantilla de producte es genera, automàticament, una variant de producte. Si una *template* només té una variant, les modificacions fetes a la instància de la plantilla es replicaran a la variant i viceversa. És, per això, que els camps particulars de cada variant es mostraran sempre quan es consulte una variant de producte però només quan es consulte la plantilla de producte si aquesta

té sols una variant (és a dir, la variant creada automàticament). En el cas que la plantilla tinga més d'una variant, els camps particulars de cada variant es mostren quan es consulta cadascuna de les variants però s'oculten quan es consulta la *template*.

El present desenvolupament haurà de respectar aquestes premisses. El resultat ha de ser, per tant, el següent:

- Camps generals a totes les variants d'una plantilla. Es mostren 3 checkbox per tal de seleccionar si es tracta d'un PG (camp `is_generic`), d'un PVM (camp `is_pvm`) o d'un PVC (camp `has_conversion_rate`). Només s'ha de poder seleccionar una de les opcions. Així mateix, si se selecciona l'opció PG, es mostra un desplegable per triar el subtipus de PG (atribut `generic_units`). Aquests camps han de ser només lectura a les variants d'una *template* amb diverses variants.
- Camps particulars per a les variants Aquests camps són configurables en cadascuna de les variants, però s'han de mostrar a la plantilla quan aquesta només disposa d'una variant (és a dir, de la variant creada automàticament amb la creació de la plantilla). En cas que la *template* tinga més d'una variant, aquests camps s'ocultaran i es mostrarà un missatge per indicar que la configuració d'eixos atributs es realitza des de cadascuna de les variants. Els camps afectats són els següents:
 - `conversion_rate`. Només es mostra si a la plantilla s'indica que és un PVC.
 - `use_default_dimensions`. Només es mostra si el producte és PG.
 - `width`, `height`, `item_length`. Es corresponen amb les dimensions que s'utilitzen per defecte quan es trie el PG. Només apareixen si l'opció `use_default_dimensions` està activa.
 - `block_default_dimensions`. Només es mostra si l'opció `use_default_dimensions` està seleccionada.

sale.view_order_form (vista heretada)

Aquesta és la vista formulari de les vendes, que haurà de mostrar el volum de fusta venuda (camp `m3_sold`) a través de tots els productes inclosos a les línies de venda. Aquest camp serà de només lectura. A les línies de venda s'han d'afegir els següents atributs de `sale.order.line`:

- `width`, `height` i `item_length`. Només visibles si el producte és PG.
- `m3`, `m2` i `ml`. El primer d'ells serà visible si el producte és PG, PVC o PVM. El segon es visualitzarà si el producte és PG de tipus `m2` i el tercer es mostrarà si el producte és PG de tipus `ml` o `pal`. `m2` i `ml` sempre seran només de lectura, mentre que `m3` es podrà editar si el producte és de tipus PVM.
- `m3_sold`. Fusta venuda en cadascuna de les línies de venda. Només serà visible si el producte és PG, PVM o PVC. És un camp de només lectura.

stock.view_picking_form (vista heretada)

Aquesta vista heretada mostra els albarans i ha d'incorporar el camp m3_sold per mostrar la fusta total de les línies incloses. Cadascuna de les línies ha d'incloure els mateixos camps que les línies de venda però del model stock.move.

stock.view_stock_move_line_detailed_operation_tree (vista heretada)

Aquesta vista heretada mostra les línies detallades d'albarà. Cadascuna d'aquestes línies ha d'incloure els mateixos camps que les línies d'albarà però del model stock.move.line.

account.view_move_form (vista heretada)

Aquesta vista heretada s'empra per a mostrar les factures. Aquesta vista haurà d'incorporar els mateixos camps que la vista de les vendes però dels models account.move i account.move.line.

5.4.2.4. Data

Caldrà crear un fitxer de dades amb instàncies predefinides del model decimal.precision. Aquest model permet indicar quantes xifres decimals han d'utilitzar els camps de tipus Float que tenen aquesta precisió especificada. Amb la definició d'aquestes instàncies en un fitxer de dades xml s'aconsegueix que, en instal·lar el mòdul, es generen automàticament al sistema. Addicionalment caldrà indicar als paràmetres de la definició dels camps float afectats quina precisió decimal s'aplica.

A l'apartat 4.3.4 s'indicaven com a part dels requisits el número de decimals que alguns atributs de tipus Float havien de tindre.

5.4.3. Generic Products Price

En aquest apartat es detallen els models, operacions, atributs i vistes, tant noves com heretades, que s'utilitzaran al mòdul "Generic Products Price".

5.4.3.1. Atributs

sale.order.line - Nous camps del model heretat

Nom	Tipus	Funció
secondary_unit_price	Float	Preu per unitat secundària d'un PG.
price_found	Boolean	Preu del PG trobat a una matriu de preus.

account.move.line - Nous camps del model heretat

Nom	Tipus	Funció
secondary_unit_price	Float	Preu per unitat secundària d'un PG.
price_found	Boolean	Preu del PG trobat a una matriu de preus.

generic.product.pricelist¹⁷ - Nou model

Nom	Tipus	Funció
name	Char	Nom de la pricelist.
product_ids	Many2many	Relació amb els product.product als quals se'ls aplica la tarifa.
pricelist_ids	Many2many	Relació amb les <i>pricelists</i> base d'Odoo, de manera que quan alguna de les <i>pricelists</i> definides en aquest camp se seleccionen a una venda o factura s'aplicaran els preus definits a la generic.product.pricelist.
generic_product_pricelist_line_ids	One2many	Relació de les línies generic.product.pricelist.line d'aquesta generic.product.pricelist.
secondary_units	Char	Unitat secundària dels productes en què s'aplica la generic.product.pricelist.

¹⁷ El model generic.product.pricelist serveixen per definir les matrius de preus dels PG

generic.product.pricelist.line¹⁸

Nom	Tipus	Funció
item_length	Float	Longitud en metres.
width	Float	Ample en centímetres.
height	Float	Gruix en centímetres.
secondary_unit_price	Float	Preu per unitat secundària.
unit_price	Float	Preu unitari d'una unitat de producte amb les dimensions especificades.
generic_product_pricelist_id	Many2one	Relació amb la generic.product.pricelist a la qual pertany la generic.product.pricelist.line.

Quan se seleccione a una venda o factura un PG i introduïsquen les dimensions es comprovarà si, amb la tarifa seleccionada, existeix alguna generic.product.pricelist relacionada amb el producte triat. Si és així i, a més a més, existeix alguna generic.product.pricelist.line que faci referència a les dimensions especificades, s'haurà de mostrar tant el preu per unitat secundària com el preu unitari calculat a partir del preu per unitat secundària i les dimensions del PG.

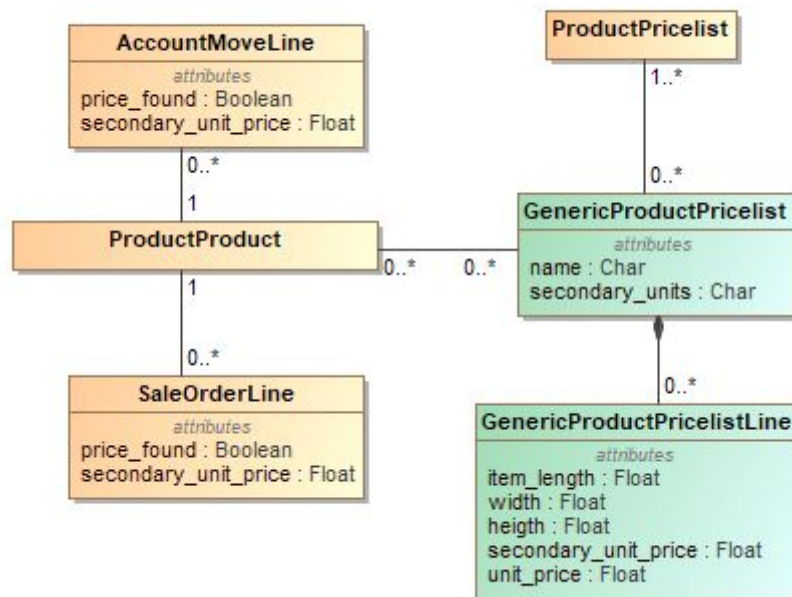


Diagrama 15

¹⁸ El model generic.product.pricelist.line serveixen per definir les línies de les matrius de preus dels PG.

5.4.3.2. Operacions i restriccions

Definir el preu dels PG segons les seues dimensions i el preu per unitat secundària a les matrius de preus de PG

Operacions

generic.product.pricelist

```
@api.depends('products_ids')
def _compute_secondary_units(self):
```

Calcula amb quina unitat secundària es relaciona la matriu (tenint en compte que els productes de tipus pal utilitzen com a unitat secundària els metres lineals). Actualitza el valor del camp `secondary_units` amb el resultat d'aquest càlcul.

```
@api.onchange('secondary_units')
def onchange_secondary_units_products_domain(self)
```

Quan canvia la unitat secundària dels productes amb què s'utilitza la matriu, cal recalculer el domini de productes seleccionables. És a dir, en el moment en què es vol crear una nova matriu de preus de PG es mostren com a seleccionables tots els PG. Una vegada se selecciona el primer producte i es calcule la unitat secundària que utilitza la matriu (funció `_compute_secondary_units`) el domini dels productes seleccionables s'actualitzarà per mostrar només els PG amb la unitat secundària corresponent a la *pricelist*. Aquesta funció, per tant, retorna el domini dels productes seleccionables.

```
def get_price(self, length=0.0, width=0.0, height=0.0)
```

Comprova si a la matriu existeix alguna línia amb les dimensions introduïdes. Retorna la línia de la matriu i el preu unitari segons les dimensions passades com a paràmetre en cas que es troben en alguna de les línies de la matriu de preus.

generic.product.pricelist.line

```
@api.depends('item_length', 'width', 'height', 'secondary_unit_price',
'generic_product_pricelist.secondary_units')
def _compute_price(self):
```

Computa el preu unitari d'una línia de la tarifa de PG a partir del preu per unitat secundària introduït i els metres cúbics, metres quadrats o metres lineals resultants de les dimensions introduïdes a la línia. Actualitza el valor de `unit_price` de la línia de la matriu de preus amb el resultat.

Restriccions

generic.product.pricelist

```
@api.constrains('product_ids', 'pricelist_ids')
def _check_unique_product(self):
```

Cada PG només pot estar present a una matriu de productes genèrics relacionada amb una *pricelist*. En cas que aquesta restricció no es complisca, es mostrarà un missatge d'error i no es permetrà guardar la matriu de preus.

generic.product.pricelist.line

```
@api.constrains('item_length', 'width', 'height')
def _check_unique_dimensions(self)
```

A cada matriu de PG només pot haver-hi una única línia amb una combinació de dimensions determinada. En cas que no es complisca, es mostra un missatge avisant que hi ha línies duplicades i no es permet guardar la matriu.

```
@api.constrains('width', 'height')
def _check_stick_dimensions(self)
```

Si a la matriu existeixen PG de tipus pal, es comprova que a les línies de la matriu el valor de l'atribut *width* siga més gran que *height*. En cas que la restricció no es complisca, es mostra un missatge que indica que a les matrius amb PG de tipus pal totes les línies han de complir la condició *ample > gruix*.

Calcular el preu per unitat secundària a vendes i factures.

sale.order.line, account.move.line

```
@api.depends("price_unit")
def _compute_secondary_unit_price(self):
```

Calcula el preu per unitat secundària dels PG, fent la divisió del subtotal de la línia entre els m3, m2 o ml segons corresponga per la tipologia del producte. Actualitza el valor de *price_sec_unit* amb el resultat obtingut de la divisió anterior.

Calcular el preu unitari a partir del preu per unitat secundària a vendes i factures

sale.order.line, account.move.line

```
@api.onchange('secondary_unit_price', 'm3', 'm2', 'ml')
def _onchange_secondary_unit_price(self):
```

Actualitza el valor de price_unit segons el valor introduït com a preu per unitat secundària.

Mostrar a les vendes el preu per unitat secundària i el preu unitari d'un PG quan s'introdueixen dimensions presents a una matriu de preus

sale.order.line

```
@api.onchange('item_length', 'width', 'height')
def generic_product_dimensions_change(self):
```

En produir-se canvis en les dimensions de la línia, es crida a la funció de la superclasse product_id_change des de la funció generic_product_dimensions_change. Aquesta funció és la que es llança quan es canvia el producte de la línia i ho fa automàticament amb l'ús de decorador onchange. La funció product_id_change, entre altres coses, actualitza el camp de la línia que conté el preu unitari i, amb ell, en cascada, s'actualitzen la resta de camps relacionats amb el preu. Per actualitzar el preu unitari crida a la funció _get_display_price, la qual també s'ha d'heretar per poder mostrar el preu per unitat secundària i el preu unitari d'un PG quan s'introdueixen dimensions presents a una matriu de preu. A continuació s'explica aquesta funció.

```
@api.depends('product_id')
def _get_display_price(self, product):
```

Funció heretada que determina el preu del producte. En el cas d'aquest desenvolupament s'haurà de tindre en compte una nova casuística: que el producte siga PG i comprovar si existeix una matriu de preus amb les dimensions introduïdes. En cas que així siga, es retornarà el preu unitari present a la matriu. A més a més, en cas que es trobe el preu a una matriu, s'actualitza el camp price_found a valor vertader.

Mostrar a les factures el preu per unitat secundària i el preu unitari d'un PG quan s'introdueixen dimensions presents a una matriu de preus

La raó per la qual les funcions per aconseguir aquest objectiu no poden ser les mateixes a les vendes i factures és perquè cada model té uns mecanismes diferents per als càlculs dels preus unitaris.

account.move.line

```
@api.depends('product_id', 'item_length', 'width', 'height')
def _compute_price_found(self)
```

En canviar el producte o les dimensions, es busca una línia de matriu de preus relacionada amb el producte i amb les dimensions introduïdes. En cas que es trobe, el valor de `price_found` es posa com a vertader.

```
@api.onchange('product_id', 'item_length', 'width', 'height')
def _generic_price_unit(self)
```

Aquesta funció s'executa quan canvien les dimensions o el producte, busca una matriu de preus amb els valors introduïts i actualitza el valor del preu unitari.

5.4.3.3. Vistes

generic_product_price.generic_product_pricelist_form_view (vista nova)

Aquesta vista mostra les matrius de preus dels PG. Del model `generic.product.pricelist` conté els següents camps:

- `name`. Nom de la matriu.
- `product_ids`. Relació dels productes als quals s'aplica la matriu.
- `pricelist_ids`. Les tarifes amb les quals s'utilitza la matriu.
- `secondary_units`. Camp només de lectura amb la unitat secundària a què s'aplica la matriu, segons els productes seleccionats.

Cada fila de la matriu conté els següents camps del model `generic.product.pricelist.line`:

- `width`, `height` i `item_length`. Camps amb les dimensions en què s'aplica el preu de la línia.
- `secondary_unit_price`. El preu per unitat secundària.
- `unit_price`. Camp de només lectura amb el resultat de calcular el preu unitari segons les dimensions introduïdes i el preu per unitat secundària establert.

A la Captura 15 es mostra aquesta vista, ja desenvolupada.

A

Products: Biga
Pricelists: Public Pricelist (EUR)
Secondary Units: m3

Generic Product Prices

Width	Height	Length	Price per Secondary Unit	Unit Price
20.00	20.00	2.00	300.00	24.00
30.00	20.00	2.00	350.00	42.00

Captura 15

sale.view_order_form (vista heretada)

La vista de les vendes ha d'incorporar, en cadascuna de les línies, el preu per unitat secundària (camp price_sec_unit). Aquest camp només serà visible si el producte és de tipus PG (indicat pel camp show_dimensions introduït per l mòdul "Generic Products and Used Wood").

account.view_move_form (vista heretada)

A la vista de les factures s'ha d'afegir, com a les vendes, un camp que mostre el preu per unitat secundària (price_sec_unit) i que siga visible si el producte és de tipus PG (indicat pel camp show_dimensions introduït pel mòdul "Generic Products and Used Wood").

5.4.4. PEFC Instances

En aquest apartat s'analitzen els atributs dels models emprats al mòdul "PEFC instances", així com les seues operacions i vistes.

5.4.4.1. Atributs

sale.order - Nous camps del model heretat

Nom	Tipus	Funció
is_pefc	Boolean	La venda utilitza fusta PEFC.

stock.picking - Nous camps del model heretat

Nom	Tipus	Funció
is_pefc	Boolean	L'albarà utilitza fusta PEFC.

account.move - Nous camps del model heretat

Nom	Tipus	Funció
is_pefc	Boolean	La factura utilitza fusta PEFC.

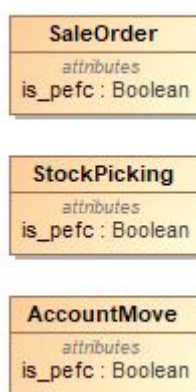


Diagrama 16

5.4.4.2. Operacions i restriccions

Passar si es tracta d'una venda amb fusta PEFC a l'albarà

sale.order

```
def action_confirm(self):
```

Mètode heretat de la superclasse que es crida quan es confirma una venda i, en conseqüència, es generen els albarans. La funció actualitza el camp `is_pefc` dels albarans generats segons si la venda utilitza o no fusta PEFC.

Passar si es tracta d'una venda amb fusta PEFC a una factura

Operacions

sale.order

```
def _prepare_invoice(self):
```

Mètode heretat de la superclasse. Quan es crida el mètode de la superclasse, es rep un diccionari amb els valors que cal passar de la venda a la factura. Caldrà actualitzar aquest diccionari amb el valor del camp `is_pefc` de la venda i retornar el diccionari amb aquesta informació addicional.

5.4.4.3. Vistes

sale.view_order_form (vista heretada)

La vista de les vendes ha d'incorporar un *checkbox* per indicar si la venda utilitza fusta PEFC (atribut `is_pefc`).

stock.view_picking_form (vista heretada)

La vista dels albarans ha d'incorporar un *checkbox* per indicar si s'utilitza fusta PEFC (atribut `is_pefc`).

account.view_move_form (vista heretada)

La vista de les factures ha d'incorporar un *checkbox* per indicar si s'utilitza fusta PEFC (atribut `is_pefc`).

5.4.5. Treatments Application

A continuació es presenten les classes emprades al mòdul “Treatments Application” amb els seus atributs, mètodes, restriccions i les vistes utilitzades.

5.4.5.1. Atributs

product.product - Nous camps del model heretat

Nom	Tipus	Funció
is_treatment	Boolean	El producte és un tractament.
allowed_treatment_ids	Many2many	Relació amb instàncies de product.product tractaments per indicar tractaments aplicables.

sale.order.line - Nous camps del model heretat

Nom	Tipus	Funció
is_treatment_line	Boolean	La línia és un tractament.
is_treating	Many2one	Relació d'una línia de tractament cap a la línia que tracta.
treatment_ids	One2many	Relació d'una línia cap a les línies que representen els seus tractaments.
can_be_treated	Boolean	La línia conté un producte que accepta tractaments.

sale.order - Nous camps del model heretat

Nom	Tipus	Funció
treatment_ids	One2many	Conjunt de línies de la venda que són tractaments.

stock.move - Nous camps del model heretat

Nom	Tipus	Funció
treatments_description	Char	S'indiquen els tractaments aplicats al producte d'una línia d'albarà.

stock.move.line - Nous camps del model heretat

Nom	Tipus	Funció
treatments_description	Char	S'indiquen els tractaments aplicats al producte d'una línia detallada d'albarà.

Els tractaments no s'afegeixen als albarans com sí ocorre a línies de venda per una raó: els tractaments es gestionen com a productes de tipus serveis i no com a productes de tipus emmagatzemables o consumibles. Els productes de tipus servei no generen línies a l'albarà i, per tant, no es poden incloure. És per això que apareixerà a cada línia de l'albarà un camp de text amb l'enumeració dels tractaments aplicats (camp treatments_description).

account.move.line - Nous camps del model heretat

Nom	Tipus	Funció
is_treatment_line	Boolean	La línia és un tractament d'una altra línia.
is_treating	Many2one	Relació d'una línia de tractament cap a la línia que tracta.
treatment_ids	One2many	Relació d'una línia cap a les línies que representen els seus tractaments.
can_be_treated	Boolean	La línia conté un producte que accepta tractaments.

account.move - Nous camps del model heretat

Nom	Tipus	Funció
treatment_ids	One2many	Línies de la factura que són tractaments.

product.treatment.wizard - Nou model

Es tracta d'un assistent que permet seleccionar a les vendes quins tractaments s'apliquen als productes que n'accepten, així com seleccionar el preu de cadascun dels tractaments.

Nom	Tipus	Funció
treatment_ids	Many2many	Relació de tots els productes-tractament del sistema.
order_line_ids	Many2many	Línies de la venda a les quals se'ls apliquen els tractaments.
price	Float	Preu del tractament.

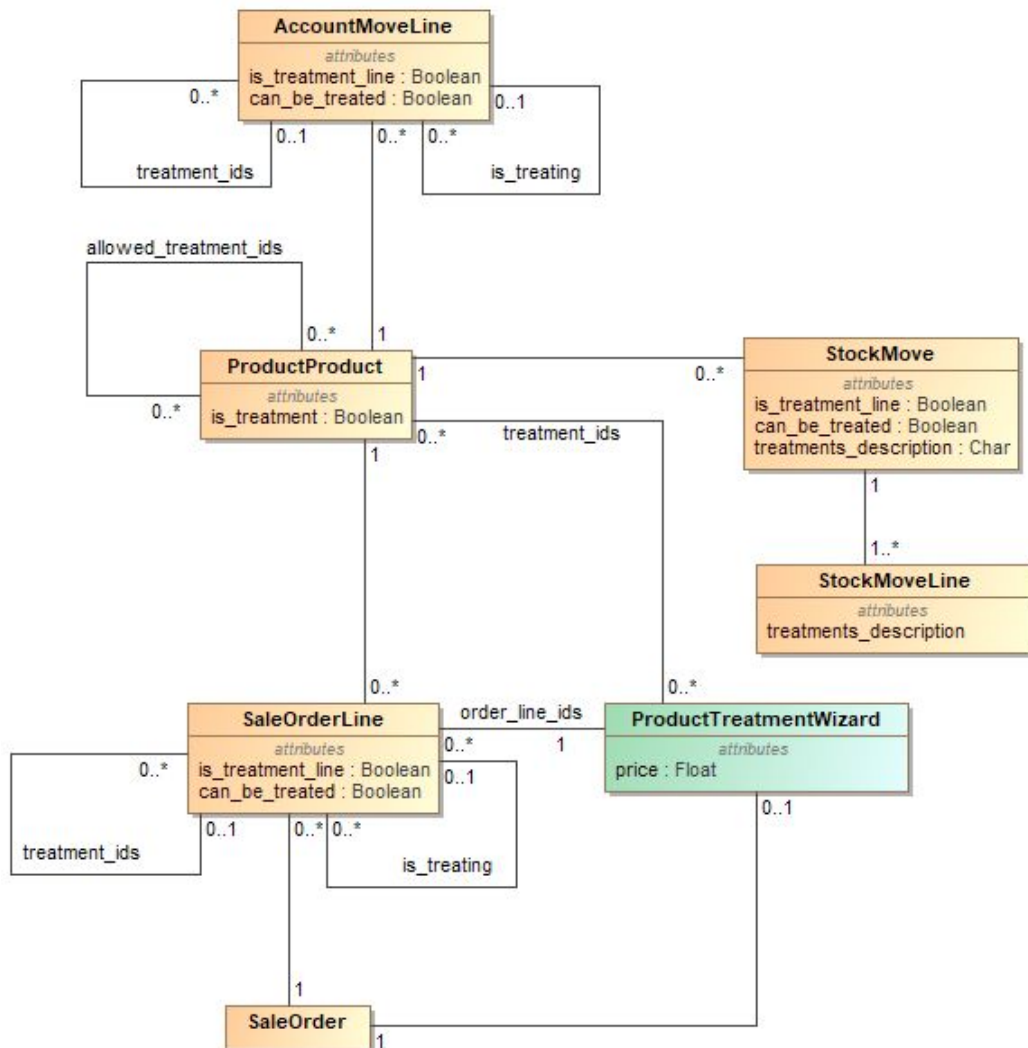


Diagrama 17

5.4.5.2. Operacions i restriccions

Definir un producte com a tractament

Restriccions

product.template

```
@api.constrains('is_treatment', 'type')
def _check_is_treatment(self):
```

Com que els tractaments han de ser productes de tipus servei, es comprova que no es defineix un producte com a tractament que no siga de tipus servei.

Seleccionar en una venda a través d'un assistent quins tractaments s'apliquen als productes presents a les línies i establiment del preu del tractament

Operacions

product.treatment.wizard

```
@api.onchange('treatments_list')
def _domain_order_lines(self):
```

Actualitza el domini de les línies que es mostren a l'assistent per aplicar els tractaments segons els tractaments triats. És a dir: cada vegada que se seleccionen tractaments des de l'assistent, s'actualitzarà la llista de línies de venda a les quals se'ls pot aplicar el tractament perquè només es puguin aplicar els tractaments triats a les línies els productes de les quals poden rebre eixos tractaments.

Aplicar els tractaments seleccionats des de l'assistent a les línies de venda

Operacions

product.treatment.wizard

```
def set_attributes(self):
```

Mètode que es crida des de l'assistent quan aquest s'accepta. Vincula els tractaments seleccionats a les línies de la venda triades. Per tal que les línies de tractament queden just sota la línia de venda a la qual tracten s'ha de tindre en compte el camp 'sequence' de les

línies de venda, el qual indica l'ordre en què apareixen les línies de venda a la pantalla. Per exemple, abans de l'aplicació de tractaments, la venda conté les següents línies:

Sequence	Producte
1	Biga
2	Troncs

En aplicar els tractaments 'Pintura' i 'Poliment' al producte 'Biga', les línies de venda haurien de quedar de la següent manera:

Sequence	Producte
1	Biga
2	Pintura
3	Poliment
4	Troncs

És a dir, la seqüència dels tractaments és contigua al producte al qual tracten.

sale.order

```
@api.depends('order_line')  
def _compute_treatment_ids(self):
```

Identifica totes les línies de la venda que contenen productes de tipus tractament i actualitza el camp `treatment_ids` amb aquesta informació.

```
def reorder_lines(self)
```

Funció que es crida quan es produeix qualsevol operació d'escriptura a la venda o quan aquesta es valida per tal de reordenar les seqüències de les línies amb l'objectiu que els tractaments sempre apareguen sota les línies a les quals tracten.

```
def write(self, values):
```

Mètode heretat de la superclasse, encarregat de realitzar una operació d'escriptura a la venda amb els valors que rep del paràmetre 'values', un diccionari amb els valors que s'han d'escriure. S'aprofita aquest mètode per a identificar quines línies de productes tractats s'eliminen per a eliminar, juntament amb elles, les línies dels tractaments aplicats. Així mateix, en llançar-se la funció `write` també es llançarà la funció `reorder_lines`.

```
def action_confirm(self):
```

Mètode heretat que, en cridar-se, genera l'albarà i permet crear una factura a partir de la venda. S'aprofita l'herència d'aquest mètode per cridar la funció `reorder_lines` i assegurar-se que les línies de tractament apareixen sota les línies a les quals tracten.

sale.order.line

```
@api.depends('product_id')
```

```
def _compute_can_be_treated(self):
```

Es comprova si el producte accepta algun tractament i actualitza el camp `can_be_treated` segons aquesta informació.

```
def _prepare_procurement_values(self):
```

Mètode heretat. En cridar el mètode de la superclasse, es rep un diccionari amb els valors de les línies de la venda que cal passar a les línies d'albarà. Cal actualitzar aquest diccionari amb la denominació que s'ha de mostrar a la línia d'albarà en el camp `treatments_description`, que és el resultat de concatenar els tractaments aplicats. És important mostrar a l'albarà els tractaments aplicats, ja que les línies de tractament de la venda, en tractar-se de línies de tipus servei, no es passen de la venda a l'albarà.

stock.rule

```
_get_custom_move_field(self):
```

Mètode heretat que recull el nom dels camps que cal passar als albarans que no formen part d'Odoo base. En primer lloc, cal cridar al mètode de la superclasse (`stock.rule`) per rebre una llista amb els camps i, a continuació, s'ha d'afegir el camp `treatments_description`.

stock.move

```
@api.depends('sale_line_id.treatment_ids')
```

```
def _compute_treatments_description(self):
```

En cas que canvien els tractaments aplicats a les línies de venda relacionades amb les línies d'albarà, es recalcula el camp `treatments_description` de la línia d'albarà.

```
def _prepare_move_line_vals(self):
```

Funció heretada que passa els valors de la línia de l'albarà a les línies detallades de l'albarà. En cridar a la funció de la superclasse es rep un diccionari amb els valors que es passen de les línies d'albarà a les línies detallades d'albarà. Cal actualitzar aquest diccionari amb el valor de `treatments_description`.

account.move

```
@api.depends('sale_line_ids')
def _compute_is_treating(self):
```

Aquesta funció identifica si les línies de factura tenen relacionades línies de venda i, en cas afirmatiu, comprova si aquestes línies són tractaments. Si són tractaments, relaciona dins la factura la línia de tractament amb la línia que tracta.

Restriccions

sale.order.line, account.move.line

```
@api.constrains('is_treating_id')
def _check_can_treat(self):
```

S'assegura que només els productes definits com a tractaments tracten altres productes.

```
@api.constrains('treatment_ids')
def _check_can_be_treated(self):
```

S'assegura que només els productes tractats poden ser, efectivament, tractats pels tractaments seleccionats.

5.4.5.3. Vistes

product.product_template_form_view (vista heretada)

Aquesta vista heretada mostra les dades de la plantilla de producte. Haurà d'ampliar-se amb els següents camps:

- `is_treatment`. Checkbox per indicar que el producte és un tractament. Serà només visible si s'ha seleccionat que el producte és de tipus servei.
- `allowed_treatments`. Seleccionar quins tractaments es poden aplicar al producte. Es mostrarà si el producte no és de tipus servei.

sale.view_order_form (vista heretada)

Vista heretada amb informació de la venda. Ha d'incloure la següent actualització:

- S'ha d'afegir un element decorador a les línies que siguen tractaments perquè es mostren de color diferent (quan el camp `is_treatment` siga vertader).
- S'ha d'incloure un botó sobre les línies de venda per obrir l'assistent que permet afegir tractaments als productes.

stock.view_picking_form (vista heretada)

Vista heretada dels albarans. A les línies, s'ha d'incorporar una nova columna amb el camp `treatments_description` (tant de les línies d'albarà com de les línies detallades d'albarà), que contindrà un text amb la relació dels tractaments aplicats al producte de la línia.

account.view_move_form (vista heretada)

Vista de les factures. De la mateixa manera que la vista de vendes, ha d'incorporar un decorador que canvie les línies de color quan el producte que continguin siga un tractament (quan el camp `is_treatment` siga vertader).

product_treatment.product_treatment_wizard_view (vista nova)

Vista de l'assistent d'incorporació de tractaments als productes. Ha de contindre els següents camps i elements:

- `treatments_list`. Relació dels tractaments que es desitja aplicar.
- `price`. Preu dels tractaments que s'apliquen.
- Les diferents línies de la venda a les quals se'ls pot aplicar un tractament. Cadascuna de les línies disposa d'un *checkbox* que es pot marcar per indicar a quines línies s'han d'aplicar els tractaments.
- Botó d'acceptar. En polsar-ho, s'apliquen els tractaments triats als productes seleccionats.
- Botó cancel·lar. Es tanca l'assistent sense realitzar cap acció.

Al Prototip 1 es mostra esquemàticament aquesta vista.

Aplicar Tractaments

Tractaments	<input type="text" value="TR1 TR2"/>			
Preu	<input type="text" value="5,5"/>			
	Producte	Llarg	Ample	Gruix
<input type="checkbox"/>	Barilla	2	5	1
<input type="checkbox"/>	Tauler	1,5	60	3
<input type="button" value="ACCEPTAR"/>		<input type="button" value="CANCEL·LAR"/>		

Prototip 1

5.4.6. New Information Reports

5.4.6.1. Atributs

sale.order - Nous camps del model heretat

Nom	Tipus	Funció
show_volume	Boolean	Mostrar la columna de fusta emprada als informes.
show_dimensions	Boolean	Mostrar la columna de dimensions als informes.

stock.picking - Nous camps del model heretat

Nom	Tipus	Funció
show_dimensions_move	Boolean	Mostrar les columnes de dimensions a les línies stock.move.
show_dimensions_move_lines	Boolean	Mostrar les columnes de dimensions a les línies stock.move.line.
show_volume_move	Boolean	Mostrar la columna de fusta emprada als informes amb línies stock.move.
show_volume_move_lines	Boolean	Mostrar la columna de fusta emprada als informes amb línies stock.move.line.

account.move - Nous camps del model heretat

Nom	Tipus	Funció
show_volume	Boolean	Mostrar la columna de fusta emprada als informes.
show_dimensions	Boolean	Mostrar la columna de dimensions als informes.

res.company - Nous camps del model heretat

Aquest model conté informació general sobre la companyia que utilitza l'ERP.

Nom	Tipus	Funció
footer_text	Text	Text que s'ha de mostrar al peu dels informes en PDF en cas que s'utilitze fusta PEFC.
footer_image	Binary	Imatge que s'ha de mostrar al peu dels informes en PDF en cas que s'utilitze fusta PEFC.
pefc_paperformat	Many2one	Format de paper que s'ha d'emprar quan es generen informes PDF de productes de fusta PEFC.

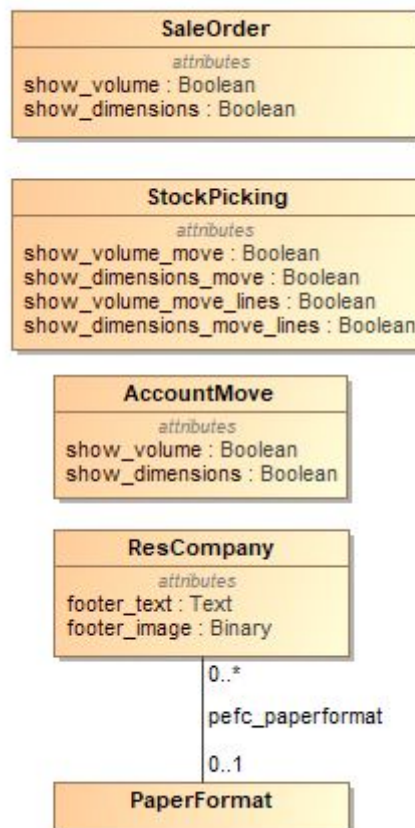


Diagrama 18

5.4.6.2. Operacions i restriccions

Operacions

sale.order

```
def _compute_show_volume(self):
```

Si existeix alguna línia de venda amb el valor de camp `show_volume` establert a vertader, es posa el valor del camp `show_volume` de `sale.order` com a vertader per indicar que cal mostrar als informes PDF de vendes la columna de fusta emprada.

```
def _compute_show_dimensions(self):
```

Si existeix alguna línia de venda que conté un PG, el valor del camp `show_dimensions` de la venda es posa en vertader per indicar que cal mostrar als informes PDF les columnes de dimensions i d'unitats secundàries de les línies.

stock.picking

```
def _compute_show_volume_move(self):
```

Si existeix alguna línia de l'albarà amb el valor de camp `show_volume` establert a vertader, es posa el valor del camp `show_volume_move` de `stock.picking` com a vertader per indicar que cal mostrar als informes PDF d'albarà la columna de fusta emprada.

```
def _compute_show_volume_move_lines(self):
```

Si existeix alguna línia detallada de l'albarà amb el valor de camp `show_volume` establert a vertader, es posa el valor del camp `show_volume_move_lines` de `stock.picking` com a vertader per indicar que cal mostrar als informes PDF d'albarà detallat la columna de fusta emprada.

```
def _compute_show_dimensions_move(self):
```

Si existeix alguna línia d'albarà que conté un PG, el valor del camp `show_dimensions_move` de l'albarà es posa en vertader per indicar que cal mostrar als informes PDF d'albarà les columnes de dimensions i unitats secundàries de les línies.

```
def _compute_show_dimensions_move_lines(self):
```

Si existeix alguna línia d'albarà detallat que conté un PG, el valor del camp `show_dimensions_move_lines` de l'albarà es posa en vertader per indicar que cal mostrar als informes PDF d'albarà detallat les columnes de dimensions i unitats secundàries.

account.move

```
def _compute_show_volume(self):
```

Si existeix alguna línia de factura amb el valor de camp show_volume establert a vertader, es posa el valor del camp show_volume d'account.move com a vertader per indicar que cal mostrar als informes PDF de factures la columna de fusta emprada.

```
def _compute_show_dimensions(self):
```

Si existeix alguna línia de factura que conté un PG, el valor del camp show_dimensions de la factura es posa en vertader per indicar que cal mostrar als informes PDF les columnes de dimensions i unitats secundàries de les línies.

ir.actions.report

```
def get_paperformat(self):
```

Funció heretada que retorna el format del paper que cal utilitzar per generar un PDF. A la funció heretada caldrà comprovar si les instàncies el PDF de les quals es vol generar pertanyen als models sale.order, stock.move o account.move. En cas afirmatiu i si la fusta utilitzada és PEFC, es retorna el format de paper definit al camp pefc_paperformat del model res.company.

5.4.6.3. Vistes

base.view_company_form (vista heretada)

Aquesta vista conté informació sobre l'empresa que utilitza l'ERP. Per mostrar als informes una sèrie de distintius quan s'empra fusta PEFC, s'han d'incloure els següents camps:

- footer_image. Imatge al peu de vendes, albarans i factures quan s'utilitza fusta PEFC.
- footer_text. Text al peu de vendes, albarans i factures quan s'utilitza fusta PEFC.
- pefc_paperformat. Seleccionable del format de paper que s'utilitza per generar els informes en PDF quan la fusta emprada a vendes, albarans i factures és PEFC.

5.4.6.4. Informes

sale.report_saleorder_document (informe heretat)

Informe dels documents de venda. Cadascuna de les línies de venda haurà d'incloure les següents columnes i modificacions:

- Llarg (item_length), ample (width), i gruix (height), per mostrar les dimensions dels PG. En cas que el producte de la línia no siga PG, no s'ha de mostrar cap valor. Les columnes de les dimensions només es mostren si l'atribut show_dimensions de la venda és vertader.
- Una columna Secondary Unit que mostrarà els camps m3, m2 o ml acompanyats de la unitat secundària segons el tipus de producte. Les columnes de m2 i ml es mostraran si l'atribut show_dimensions de la venda és vertader. La columna m3 es mostrarà si l'atribut show_volume de la venda és vertader.
- Secondary Unit Price. Columna amb el valor del camp de la línia de venda secondary_unit_price, que indica el preu per unitat secundària dels PG.
- Atribut d'estil a les línies de venda que s'active quan la línia siga un tractament amb l'objectiu de mostrar aquestes línies en color gris i no en negre.

stock.report_delivery_document (informe heretat)

Es tracta de l'informe de l'albarà. Cal que incloga els següents camps a cadascuna de les línies.

- Llarg (camp item_length), ample (camp width), i gruix (camp height). Aquests camps mostraran les dimensions dels PG. En cas que el producte de la línia no siga PG, no s'ha de mostrar cap valor. Les columnes de les dimensions només es mostraran si l'atribut show_dimensions de l'albarà és vertader.
- Una columna Secondary Unit que mostrarà els camps m3, m2 o ml acompanyats de la unitat secundària segons el tipus de producte. Les columnes de m2 i ml es mostraran si l'atribut show_dimensions de l'albarà és vertader. La columna m3 es mostrarà si l'atribut show_volume de l'albarà és vertader.
- Columna 'Treatments', amb una cadena de text amb els tractaments incorporats al producte.

stock.report_picking (informe heretat)

Aquest informe, que s'utilitza quan es genera un PDF amb les línies detalls de l'albarà, haurà d'ampliar-se amb la mateixa informació que la incorporada a stock.report_delivery_document.

account.report_invoice_document (informe heretat)

Aquesta és la plantilla d'informes de factures. Haurà d'incorporar els mateixos camps i informació que sale.report_saleorder_document amb la informació de les factures.

web.external_layout_standard (informe heretat)

Aquesta *template* conté l'encapçalament i el peu de pàgina dels documents PDF. Cal modificar-lo per tal d'afegir la imatge i el text triat per als documents de venda, factura i albarà quan la fusta emprada és de tipus PEFC.

6. Implementació i proves

6.1. Implementació

Una vegada finalitzada l'etapa de disseny, pot començar la implementació dels diferents mòduls que conformen el projecte global. D'una banda, s'ha parlat que, per les característiques del projecte, es produeix un cert acoblament entre els diferents mòduls. D'altra, les proves no es realitzaran una vegada estiguen tots els mòduls finalitzats sinó que s'hauran de realitzar al llarg de tot el procés i tan prompte com es dispose de les primeres funcionalitats comprovables. És per això que s'ha establert que l'ordre d'implementació dels mòduls ha de ser el següent:

1. Generic Products and Used Wood
2. Generic Products Price
3. Treatments Application
4. PEFC Instances
5. New Information Reports

Com a eina de control de versions s'utilitzarà Gitlab i es crearà una branca per cadascun dels mòduls. L'aplicació de les branques a la rama màster es realitzarà una vegada el mòdul estiga finalitzat i haja passat les proves unitàries relacionades. L'acceptació de fusions a la branca màster està centralitzada i només les pot autoritzar una persona, però tots els membres de l'equip poden realitzar comentaris del codi pujat pels diferents membres.

Així mateix es treballarà en tres entorns diferents:

1. **Entorn de desenvolupament.** En les màquines de cadascun dels desenvolupadors. És en aquest entorn on es realitzarà la implementació juntament amb les proves unitàries.
2. **Entorn de proves.** En aquest entorn es realitzaran les proves de sistema, tant per part de l'equip de desenvolupament com del client. Cadascun d'aquests grups tindrà assignada una base de dades. És una còpia de l'entorn de producció.
3. **Entorn de producció.** És l'entorn en què opera el client quan comença a treballar amb el *software* desenvolupat.

En el moment de lliurament d'aquesta memòria, s'ha completat una versió beta de dos dels mòduls: "Generic Products and Used Wood" i "Generic Products Price".

6.2. Proves

A l'obra *Software Engineering Body of Knowledge*, es defineix el procés de proves de programari com “una verificació dinàmica que el programa proporciona els comportaments esperats en un conjunt finit de casos de test, correctament triats de l'usualment infinit domini d'execució”¹⁹.

A l'apartat 4.5 s'han establert una sèrie de proves d'acceptació que han servit per a la posterior fase de disseny, iniciativa presa a partir dels principis del desenvolupament conduït per proves (TDD). Així i tot, es podria anar més enllà amb la proposta de proves addicionals a diferents nivells que contribuïren a garantir la qualitat del producte lliurat.

6.2.1. Planificació de les proves

Es parla de quatre aspectes: abast, enfocament, criteris d'inici i criteris de finalització²⁰.

- **Abast.** Les proves faran un recorregut per tots els requisits recollits amb l'objectiu de garantir que s'han assolit els objectius acordats amb el client. Bona part dels tests són els que han servit com a guia de disseny i, per tant, comprenen pràcticament tots els requeriments. N'hi ha dos aspectes que, en qualsevol cas, no caldrà provar amb profunditat.
 - Seguretat. Odoo s'encarrega nativament d'aquest aspecte amb els models base. Pel que fa als nous models desenvolupats, com de moment l'únic usuari del sistema serà el propietari de l'empresa, no cal testar-lo per a veure si, efectivament, els rols i permisos estan ben definits.
 - Consistència. Odoo s'encarrega d'aspectes de consistència de base de dades.
- **Enfocament.** Les proves es realitzaran tant en entorns locals com en entorns de proves als quals tinga accés el client a través d'un navegador web. L'ús d'un entorn o un altre dependrà del tipus específic de prova, definides més avant. Pel tipus de desenvolupament, el que s'haurà de recollir és, a nivell general, un *Passed* o *Failed* de cadascuna de les proves, però també comentaris que ajuden a millorar els mòduls o a identificar els errors.
- **Criteris d'inici.** Les proves, seguint les recomanacions del SWEBOK, s'hauran d'iniciar des del principi del desenvolupament del projecte. Com és evident, algunes d'elles només podran realitzar-se una vegada finalitzada la implementació mentre que d'altres s'hauran de fer tan prompte com s'haja desenvolupat una nova funcionalitat comprovable. És imprescindible que, almenys, es realitzen les proves d'acceptació de cada mòdul una vegada aquest estiga finalitzat.

¹⁹ BOURQUE Pierre, FAIRLEY, Richard E. *SWEBOK: Guide to the Software Engineering Body of Knowledge*. IEEE, 2014. ISBN 0-7695-5166-1

²⁰ ESCUDERO, Xavier. *Qualitat del programari: tècniques de prevenció, detecció i correcció de defectes* [recurs d'aprenentatge]. Barcelona:UOC

- **Criteris de finalització de les proves.** Les proves hauran de realitzar-se fins que es passen tots els tests d'acceptació establits. Cal tindre en compte que, en introduir modificacions, pot passar que una prova que abans es passava ara ja no se supere.

6.2.2. Definició de proves concretes

Les proves concretes poden dividir-se segons dos criteris: el nivell i les tècniques²¹.

6.2.2.1. Nivell de les proves

El SWEBOK parla de diferents nivells de proves, que poden dividir-se segons si fan referència a l'objecte o a l'objectiu del test.

A la primera classificació trobem els **tests unitaris**, definits com aquells adreçats a provar el funcionament aïllat dels diferents elements comprovables individualment. Els tests d'acceptació proposats en aquesta memòria són, en la gran majoria, proves d'aquest tipus. Existeixen, així mateix, les **proves d'integració**, creades per verificar la correcta interacció entre diferents components. Com que aquest projecte s'ha dividit en mòduls, aquest tipus de proves són importants, ja que el funcionament d'alguns depèn d'uns altres. A les proves d'acceptació de l'apartat 4.5 es troben també alguns tests d'aquest tipus. Tot i que el SWEBOK explica que el TDD es basa en tests unitaris, la natura del projecte fa necessari que el desenvolupament estiga guiat a través de comprovacions que garanteixen la correcta integració dels mòduls. Per exemple, no es pot comprovar que el preu a una venda es mostra correctament quan s'han introduït unes dimensions presents a una matriu de preus sense tindre productes PG, definits en un altre mòdul. En tercer lloc, existeixen els **tests de sistema**, dirigits a provar el funcionament conjuntament. Aquest podria ser aplicable a través de la creació d'un cas d'ús genèric que provara el flux de treball del client.

Pel que fa als nivells de test segons els objectius, trobem el **testatge d'acceptació**, que consisteix a proporcionar al client un entorn perquè pugui comprovar ell mateix que es compleixen els requisits. Aquest nivell pot incloure uns altres de manera implícita també esmentats al SWEBOK, com el **testatge d'usabilitat i d'interacció persona-ordinador**, ja que el client podria informar de qualsevol aspecte relacionat amb aquests temes mentre realitza les proves. Així mateix, s'han de considerar els **tests de regressió**, definits al SWEBOK com aquells que s'utilitzen quan es fan modificacions al *software* i cal comprovar que els tests que abans es passaven continuen tenint resultats positius.

²¹ ESCUDERO, Xavier. *Qualitat del programari: tècniques de prevenció, detecció i correcció de defectes* [recurs d'aprenentatge]. Barcelona:UOC

6.2.2.2. Tècniques de proves

Existeixen dos tipus principals de proves²²:

- **Les proves de caixa blanca.** Analitzen com el *software* està dissenyat, és a dir, el codi. Aquestes proves es realitzaran en aquest projecte amb l'anàlisi del codi per part d'altres membres de l'equip de desenvolupament, aprofitant les eines de comentari proporcionades per Gitlab.
- **Les proves de caixa negra.** Estudien quins *outputs* genera el sistema segons diferents *inputs*. Aquestes proves seran realitzades tant per part dels membres de l'equip de desenvolupament a través de casos predefinitos com del client a través dels casos que ell considere representatius per al seu treball del dia a dia. De les realitzades per l'equip, es trobarien totes les definides com a proves d'acceptació, inserides dins del tipus de tests de partició en equivalències, que consisteixen a dividir el domini de possibles entrades en diferents casos, tant vàlids com invàlids, per comprovar que tot funciona de la manera esperada. Seria interessant ampliar aquestes proves amb dos tipus de tests específics:
 - Tests d'anàlisi de valors frontera. Consisteix a provar valors que es troben a la frontera dels inputs acceptats i no acceptats. Per exemple, si a les dimensions dels productes PG de tipus pal l'ample ha de ser més gran que el gruix, es pot provar un cas en què s'introdueixen el mateix valor a ambdues dimensions.
 - Tests adreçats a esbrinar errors. Consisteix a fer proves amb valors que se sospita que puguen donar problemes com, per exemple, provar què passa quan a les dimensions dels PG a una venda se li assigna el 0 com a valor.

Odoon proporciona la possibilitat de testejar el codi Python a través de la definició de funcions amb tests unitaris per a executar, d'aquesta manera, els tests de manera automatitzada. Així i tot, per una qüestió d'inversió temporal, s'ha considerat adient realitzar les proves manualment, encara que no es descarta la possibilitat de dissenyar tests unitaris automàtics en cas que es dispose de temps.

²² Bourque P., Firley, R. (2014). *SWEBOK: Guide to the Software Engineering Body of Knowledge*. IEEE. Swebok

7. Conclusions i continuïtat

En aquesta memòria ha quedat recollida bona part del procés de desenvolupament d'una adaptació de l'ERP Odoo per satisfer les necessitats particulars d'un client amb uns procediments de venda molt específics. S'ha partit des de l'anàlisi de la visió global del projecte, que va permetre determinar la viabilitat, i s'ha arribat fins a la implementació d'una part de les utilitats definides.

Després d'un estudi exhaustiu dels requeriments plantejats pel client es va optar per dividir el desenvolupament en cinc mòduls, cadascun dels quals encapsula una sèrie de funcionalitats. Una vegada completada la definició dels requisits, tant funcionals com no funcionals, es va optar per conduir la fase de disseny segons el paradigma de desenvolupament guiat per proves, que va permetre tant fixar les funcionalitats amb casos pràctics concrets com avançar una part important de la planificació de l'etapa de proves. Una vegada realitzat el disseny, s'ha aprofundit en la planificació de les proves i, paral·lelament, s'ha començat amb la implementació.

El fet de treballar sobre un *framework* tan ampli com aquest ha deixat entreveure una sèrie de punts, que han sigut essencials en l'evolució de tot el projecte:

- Una part important de les necessitats que *a priori* poden semblar que necessiten d'un desenvolupament complet personalitzat, en realitat es poden cobrir amb **opcions ja presents** al *software*. En molts d'aquests casos s'ha plantejat un dilema: resulta convenient optar per solucions ja existents per donar resposta als requisits encara que no s'ajusten al 100% al demanat a canvi d'un important estalvi de temps i la garantia d'estar treballant amb desenvolupaments ja testejats? En la gran majoria dels casos, s'ha optat per fer ús d'aquelles opcions ja existents, sempre que els requisits es continuaven complint.
- Existeixen una sèrie de **patrons** existents que cal respectar. Entre altres coses, eixa és l'única manera de garantir que el nou desenvolupament encaixa correctament amb la resta del sistema i el seu flux i, a més, fer ús de patrons ja emprats contribueix, no només a l'èxit en l'assoliment de requisits sinó també a aconseguir una alta eficiència, ja que es procedeix de maneres ja provades i acceptades per altres desenvolupadors.
- La planificació i execució de la fase de **proves** esdevé un element d'enorme importància. Els mòduls resultants d'aquest projecte estaran inserits en un entorn enorme. Serà important assegurar tant que es compleixen els requisits com que el sistema, en el seu conjunt, funciona correctament.

Com s'ha comentat a la secció introductòria, el que s'ha presentat en aquesta memòria és només una part d'un projecte més ampli: la migració a la versió 13 d'Odoo d'un client que, actualment, fa servir la versió 11. Pel que fa al que s'ha exposat, obviant les tasques de desplegament, es continuaria de la següent manera:

- Començar a realitzar **proves unitàries** dels mòduls ja desenvolupats. Aquestes proves ja han estat definides per tal de guiar el disseny. També es podrien començar a executar

proves d'integració per a verificar la correcta incorporació de les noves funcionalitats al flux d'Odoo.

- Plantejar la creació d'un **entorn de proves** al qual hi tinguera accés el client perquè pugui començar a realitzar proves de verificació a mesura que es completen els diferents mòduls. Això contribuiria a detectar problemes de manera gradual, la qual cosa ajudaria a evitar trobar-se amb la urgència de fer un gran nombre de correccions al final del projecte.
- Continuar amb el desenvolupament dels tres **mòduls restants**: “Treatments Application”, “PEFC Instances” i “New Information Reports”. L'estratègia de proves hauria de ser com l'exposada als dos punts anteriors.
- Realitzar **proves de sistema**, que comprovara el flux sencer amb casos d'ús complexos.
- Tasques de **configuració** de l'entorn de producció, just abans de començar a treballar amb els nous mòduls. Per exemple, caldria crear les matrius de preus o els productes definitius.
- En un futur, introduir **actualitzacions**. Per exemple, afegir la possibilitat d'incloure tractaments a través d'un assistent als albarans i les factures de la mateixa manera que es fa amb les vendes.
- A tot això caldria afegir-li tasques de **manteniment** una vegada el producte haja estat lliurat al client.

Annexos

1. Flux bàsic d'Odoo

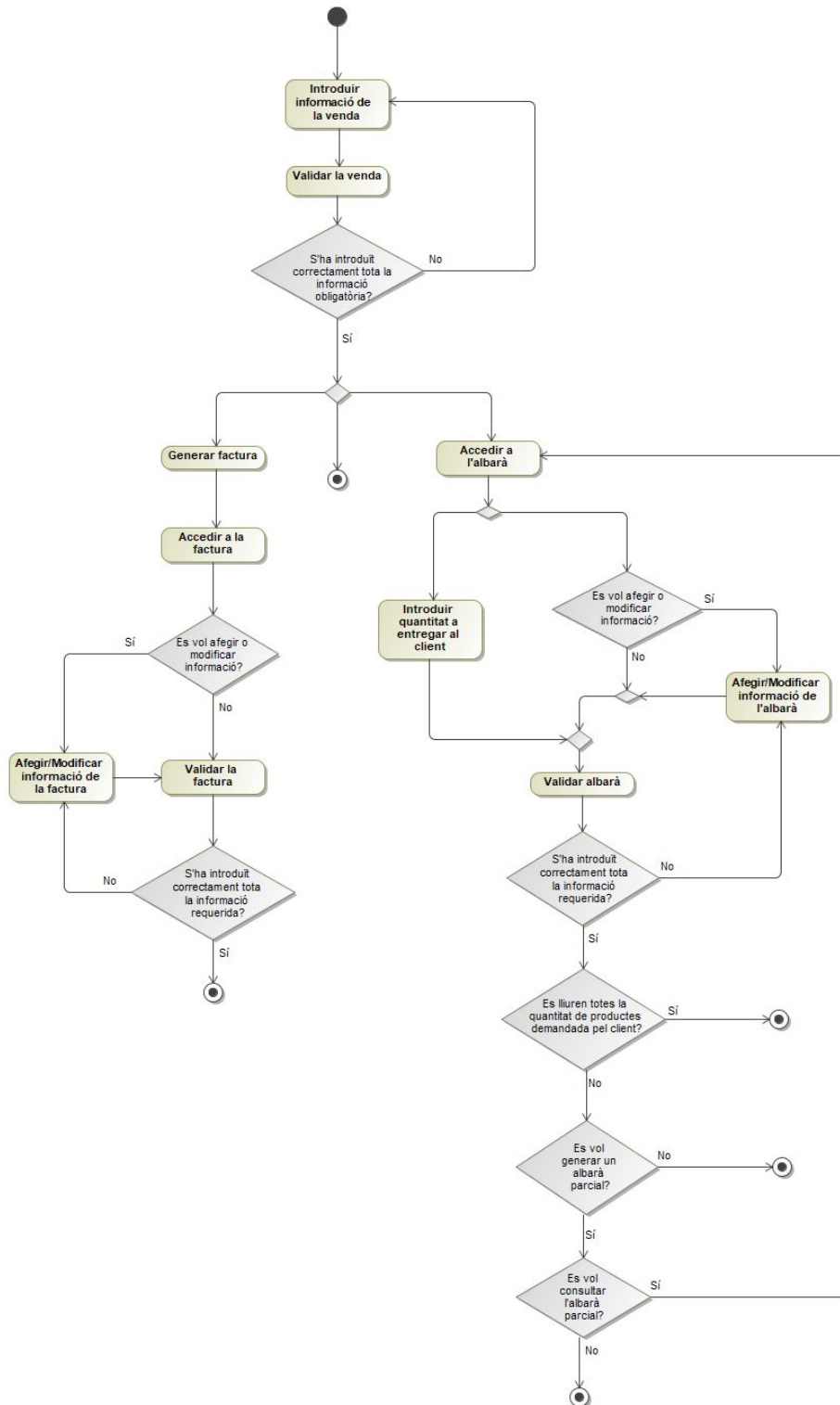


Diagrama 19

2. Casos d'ús

Cas d'ús: Generar albarà i factura a partir d'una venda

Actor principal: Gestor

Àmbit: sistema

Nivell d'objectiu: usuari

Stakeholders i interessos:

Gestor: vol enregistrar les dades d'una venda i generar un albarà i factura

Client: vol rebre un pressupost d'una venda, un albarà i una factura

Precondició: el gestor es troba a la pantalla de creació d'una venda

Garanties mínimes:

- 1) Es guarda un esborrany de la venda

Garanties en cas d'èxit:

- 1) Es genera un albarà
- 2) Es genera una factura

Escenari principal d'èxit:

- 1) El gestor selecciona al client
- 2) El sistema actualitza les dades de direcció de facturació, direcció d'enviament, plantilla de pressupost, *pricelist* i termes de pagament.
- 3) El gestor selecciona un producte i la quantitat.
- 4) El sistema actualitza la descripció de la línia
- 5) El sistema actualitza els impostos de la línia
- 6) El sistema actualitza el preu unitari de la línia
- 7) El sistema actualitza el subtotal de la línia
- 8) El sistema actualitza l'import total de la venda
- 9) El gestor valida la venda
- 10) El sistema genera un albarà i actualitza l'estat de la venda a 'Sale Order'
- 11) El gestor selecciona generar una factura a partir de la venda
- 12) El sistema genera una factura amb la informació introduïda a la venda

Extensions:

- 3a) El producte seleccionat és PG.
 - 3a1) El gestor introdueix les dimensions del producte (cas d'ús **Calcular la fusta d'un producte genèric en una venda o factura**)
 - 3a2) El sistema actualitza la fusta total emprada, el preu unitari del producte i el preu total de la línia amb el preu d'una matriu de preus de PG.
 - 3a2a) No existeix un preu associat al PG amb eixes dimensions i la pricelist seleccionada.
 - 3a2a1) El sistema dóna valor 0 al valor unitari del producte i actualitza el preu total de la línia amb valor 0.
 - 3a3) Es torna al pas 4
- 3b) El producte seleccionat és PVC.
 - 3b1) El sistema calcula el volum de fusta (cas d'ús **Calcular la fusta d'un producte amb coeficient de conversió**) i recalcula la fusta total emprada
 - 3b2) Es torna al pas 4
- 3c) El producte seleccionat és de tipus PVM
 - 3c1) El gestor introduïx el volum de fusta emprada
 - 3c2) El sistema actualitza la fusta emprada a la línia de producte i la fusta emprada total
 - 3c2) Es torna al pas 4
- 9a) El gestor vol aplicar tractaments a algun producte
 - 9a1) El gestor selecciona l'opció d'aplicació de tractaments (cas d'ús **Aplicar un tractament a un producte**)
- 9b) El gestor vol introduir més productes.
 - 9b1) Es torna al pas 3.
- 10a) No s'ha introduït correctament tota la informació obligatòria
 - 10a1) El sistema avisa al gestor que no ha introduït tota la informació obligatòria
 - 10a2) Es torna al pas 3

Cas d'ús: Validar un albarà

Actor principal: Gestor

Àmbit: sistema

Nivell d'objectiu: usuari

Stakeholders i interessos:

Gestor: vol indicar la quantitat de productes lliurats a un client i actualitzar l'estoc

Client: vol rebre informació sobre els productes que li seran lliurats

Precondició: el gestor es troba a la pantalla d'un albarà ja creat i no validat amb la informació obligatòria introduïda i les dades desitjades

Garanties mínimes: -

Garanties en cas d'èxit:

- 1) S'actualitza l'estoc amb les quantitats lliurades
- 2) Es canvia l'estat de l'albarà a 'Done'

Escenari principal d'èxit:

- 1) El gestor indica les quantitats lliurades dels productes de l'albarà del total de la quantitat demandada pel client i valida l'albarà.
- 2) El sistema passa l'albarà a l'estat 'Done'.

Extensions:

- 3a) No es lliura tota quantitat de productes demanades pel client.
 - 3a1) El sistema ofereix l'opció de generar un albarà parcial.
 - 3a1a) El gestor indica que vol generar un albarà parcial.
 - 3a1a1) El sistema crea un albarà parcial amb les quantitats de productes no lliurades dels productes demandats pel client i recalcula la fusta emprada de l'albarà base atenent a la quantitat lliurada.
 - 3a1b) El gestor indica que no vol generar un albarà parcial.
 - 3a1b1) Acaba el cas d'ús

Cas d'ús: Validar una factura

Actor principal: Gestor

Àmbit: sistema

Nivell d'objectiu: usuari

Stakeholders i interessos:

Gestor: vol formalitzar una transacció econòmica i passar la informació als llibres comptables

Client: vol rebre una factura amb les dades de la transacció

Precondició: el gestor es troba a la pantalla d'una factura ja creada i no validada amb la informació obligatòria correctament introduïda i les dades desitjades.

Garanties mínimes: -

Garanties en cas d'èxit:

- 1) Es passa la informació de la factura a la comptabilitat de l'empresa
- 2) La factura passa a l'estat 'Posted'

Escenari principal d'èxit:

- 1) El gestor valida la factura
- 2) El sistema actualitza l'estat de la factura a 'Posted' i passa la informació a la comptabilitat de la companyia

Cas d'ús: Introduir els preus de productes genèrics en una matriu de preus.

Actor principal: Gestor

Àmbit: sistema

Nivell d'objectiu: tasca

Stakeholders i interessos:

Gestor: vol enregistrar quant costen els productes genèrics segons les seues dimensions per tal d'utilitzar aquests valors a vendes i factures.

Precondició: el gestor es troba a la pantalla de creació d'una matriu de preus

Garanties mínimes: -

Garanties en cas d'èxit:

- 1) Es crea una matriu de preus de productes genèrics.

Escenari principal d'èxit:

- 1) El gestor selecciona un producte al qual aplicar la matriu de preus
- 2) El sistema actualitza la unitat secundària a la qual es pot aplicar la matriu de preus (m3, m2 o ml) i la llista de productes seleccionables, limitant-la a aquells productes genèrics que utilitzen aquesta unitat secundària
- 3) El gestor selecciona les *pricelists* a què vol aplicar la matriu de preus
- 4) El gestor crea una línia de la matriu de preus i introdueix les tres dimensions i el preu de la fusta per unitat secundària
- 5) El sistema calcula i mostra el preu unitari a partir del preu per unitat secundària i les dimensions introduïdes. (Es torna al pas 4 tantes vegades com línies es volen introduir)
- 6) L'usuari guarda la matriu
- 7) El sistema emmagatzema la matriu

Extensions:

7a) N'hi ha seleccionat algun producte de tipus pal i en alguna de les línies de la matriu s'ha indicat que el gruix és més gran que l'ample

7a1) El sistema avisa que no es compleix una restricció i es torna al pas 4

7b) Algun dels productes seleccionats ja té una matriu assignada per a alguna de les *priceslists* triades

7b1) El sistema avisa que no es compleix una restricció i es torna al pas 3

7c) Es detecta més d'una línia amb la matriu amb les mateixes dimensions

7c1) El sistema avisa que no es compleix una restricció i es torna al pas 4

Cas d'ús: Calcular la fusta d'un producte genèric en una venda, albarà o factura

Actor principal: Gestor

Àmbit: sistema

Nivell d'objectiu: tasca

Stakeholders i interessos:

Gestor: vol enregistrar la fusta que suposa un producte genèric individualment i dins d'una venda, albarà o factura

Precondició: s'ha triat un producte genèric a la línia de venda, albarà o factura

Garanties mínimes: -

Garanties en cas d'èxit:

- 1) Es calcula la fusta emprada pel producte
- 2) Es calculen els m³/m²/ml (segons producte)

Escenari principal d'èxit:

- 1) El gestor indica les dimensions del producte a la línia
- 2) El sistema calcula la fusta emprada i els m³, m² o ml segons corresponga al tipus de producte tenint en compte la quantitat seleccionada

Extensions:

- 3a) El gestor modifica alguna de les dimensions
 - 3a1) Es torna al pas 2
- 3b) El gestor modifica la quantitat
 - 3b1) Es torna al pas 2
- 3c) El gestor modifica la UoM
 - 3c1) Es torna al pas 2

Cas d'ús: Calcular la fusta d'un producte amb coeficient de conversió

Actor principal: Gestor

Àmbit: Sistema

Nivell d'objectiu: tasca

Stakeholders i interessos:

Gestor: vol enregistrar la fusta que suposa un producte amb coeficient de conversió

Precondició: s'ha triat un producte amb coeficient de conversió a la línia de venda, albarà o factura

Garanties mínimes: -

Garanties en cas d'èxit:

- 1) Es calcula la fusta emprada per eixe producte

Escenari principal d'èxit:

- 1) El sistema calcula la fusta emprada amb el coeficient definit al producte i la quantitat seleccionada..

Extensions:

- 2a) L'usuari modifica la quantitat.
 - 2a1) Es torna al pas 1
- 2b) L'usuari modifica la UoM.
 - 2b1) Es torna al pas 1

Cas d'ús: Aplicar un tractament a un producte

Actor principal: Gestor

Àmbit: sistema

Nivell d'objectiu: tasca

Stakeholders i interessos:

Gestor: vol especificar quins tractaments s'apliquen a un producte per incrementar el seu preu i saber exactament com s'ha de lliurar el producte al client.

Client: vol tindre constància de les característiques exactes dels productes i del preu que suposa l'aplicació de tractaments.

Precondició: el gestor se situa a la pantalla d'aplicació de tractaments als productes de vendes

Garanties mínimes: -

Garanties en cas d'èxit:

- 1) Es vinculen els tractaments seleccionats als productes triats i s'inclou el preu

Escenari principal d'èxit:

- 1) El gestor selecciona el tractament o tractaments que vol aplicar
- 2) El sistema actualitza la llista de productes que accepten el tractament o tractaments triats
- 3) El gestor selecciona el producte o productes als quals vol aplicar el tractament o tractaments
- 4) El gestor selecciona el preu del tractament o tractaments i accepta
- 5) El sistema vincula el tractament o tractaments al producte o productes triats

Bibliografia

Recursos d'aprenentatge

ESCUADERO, Xavier. *Qualitat del programari: gestió de la qualitat i mètriques* [recurs d'aprenentatge]. Barcelona:UOC

ESCUADERO, Xavier. *Qualitat del programari: tècniques de prevenció, detecció i correcció de defectes* [recurs d'aprenentatge]. Barcelona:UOC

PRADEL MIQUEL, Jordi i RAYA MARTOS, Jose. *Gestió de requisits* [recurs d'aprenentatge]. Barcelona: UOC

RODRÍGUEZ, Juan Ramón. *La gestió de projectes. Conceptes bàsics* [recurs d'aprenentatge]. Barcelona:UOC

Libres

BLÉ JURADO, Carlos. *Diseño Ágil con TDD*. iExpertos, 2010. ISBN: 978-1445264714

BOURQUE Pierre, FAIRLEY, Richard E. *SWEBOOK: Guide to the Software Engineering Body of Knowledge*. IEEE, 2014. ISBN 0-7695-5166-1

REIS, Daniel. *Odoo 12 Development Essentials*. Packt Publishing, 2018. ISBN 1789532477

Webs

Building a Module [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.odoo.com/documentation/13.0/howtos/backend.html>

Deploying Odoo [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.odoo.com/documentation/13.0/setup/deploy.html#supported-browsers>

ORM (Object Relational Mapping) in Odoo [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: <https://www.cybrosys.com/blog/orm-in-odoo>

Testdriven development [en línia] [data de consulta: 20 de desembre de 2020]. Disponible a: https://en.wikipedia.org/wiki/Test-driven_development

Estàndards

COMPUTER SOCIETY. *IEEE Recommended Practice for Software Requirements Specifications*. IEEE Std 830-1998. 1998

INTERNATIONAL ORGANIZATION FOR STANDARIZATION. *Software and systems engineering*. IEEE/ISO/IEC 29119. 2009